

# JAVASCRIPT IN URDU

جاوا اسکرپٹ

اردو زبان میں

☆☆☆ محمد دانش ارشاد ☆☆☆

1995 میں اپنے براوزر کے لئے بنائی تھی۔ جس کو بعد میں تمام اخیر نیٹ براوزر نے استعمال کرنا شروع کر دیا۔ ابتداء میں JavaScript کا نام LiveScript تھا جو بعد میں تبدیل کر کے JavaScript رکھ دیا گیا۔ JavaScript ایک بلکل اور آسان زبان ہے اور یہ ایک کلائنٹ سائٹ زبان ہے یعنی جس کے لئے کسی بھی قسم کی انسالیشن وغیرہ کی ضرورت نہیں ہوتی۔ ASP یا PHP وغیرہ کی طرح اس میں آپ کو کسی سرور کی ضرورت نہیں۔

آپ JavaScript کو کسی بھی نیکست ایڈیٹر، مثلاً Notepad، Notepad ++ وغیرہ میں ناچپ کر سکتے ہیں اور اپنے کسی بھی ویب براوزر میں استعمال کر سکتے ہیں۔ JavaScript ایک Case-Sensitive زبان ہے مطلب ہے کہ اس کے کسی بھی کوڈ میں جو حرف چھوٹا ہو گا تو وہ آپ کو چھوٹا ہی لکھنا ہے اور جو حرف بڑا ہو گا وہ بڑا ہی لکھنا ہے۔ JavaScript کے اندر چھوٹے اور بڑے حرف والگ الگ چیزیں اصولی کی جاتی ہیں۔

مثال کے طور پر Javascript اور javascript ان دونوں میں سے ایک کا "L" چھوٹا اور دوسرے کا بڑا ہے تو یہ دونوں الگ الگ الفاظ ہوئے۔ آپ کو اس چیز کا خیال رکھنا ہوگا۔

JavaScript میں ہم کیا کیا کر سکتے ہیں؟

JavaScript کے اندر ہم مختلف کام کر سکتے ہیں۔

(مثال):

ویب کی سیکورٹی، فارم ویلڈیشن، کوکی بنانا، یوزر کی ip، اسکا براوزر، اسکا آپرینٹنگ سسٹم کونسا ہے وغیرہ یوزر کو بتانا۔ ہم جاواسکریپٹ کے اندر گھٹری بناسکتے ہیں۔ ویب بیچ کے لئے چھوٹے موٹے گیمز وغیرہ بھی ڈیزائن کر سکتے ہیں۔

JavaScript سکھنے کے لئے کیا ضروری ہے؟

JavaScript کے لئے آپ کے پاس HTML اور CSS کی بنیادی معلومات ضروری ہے۔ اگر آپ ان کے بارے



میں نہیں جانتے تو برائے مہربانی پہلے آپ ان دونوں کو سمجھیں۔

JavaScript کے لئے آپ کو کمپیوٹر اور انٹرنیٹ کی بنیادی معلومات ہوئی ضروری ہے۔ ☆

ایک عد دنیکست ایڈیٹر جیسے: Notepad, Notepad ++ وغیرہ ہونا لازمی ہے۔ ☆

ایک عد دویب براوزر جیسے: Google Chrome, Mozilla Firefox, Internet Explorer, Safari, Opera, وغیرہ۔ ☆

JavaScript کی اس کتاب میں ہم کیا کیا سمجھیں گے؟

ہم JavaScript کی اس کتاب میں JavaScript کا بنیادی استعمال، اس کے کام کا طریقہ کار وغیرہ کے متعلق سمجھیں گے۔

## فہرست

صفحہ نمبر	باب	ترتیب
6	اور اس کا بنیادی استعمال Syntex کا JavaScript	1
8	کون کون سے ہیں؟ Methods کے JavaScript	2
8	Comments میں داخل کرنا JavaScript میں	3
9	کیا ہیں؟ Data Types کے JavaScript	4
11	کی تخلیق اور ان کا استعمال کرنا Variables میں JavaScript	5
15	کی تخلیق اور ان کا استعمال کرنا Functions میں JavaScript	6
18	کا استعمال کرنا String Method میں JavaScript	7
24	کا استعمال کرنا Number Methods میں JavaScript	8
25	کا استعمال کرنا Math Object میں JavaScript	9
28	کیا کیا ہیں اور ان کا استعمال کرنا Events میں JavaScript	10
31	کیا ہیں؟ Assignment Operators کے JavaScript	11
31	کیا ہیں؟ Arithmetic Operator کے JavaScript	12
37	کیا ہیں؟ Comparison Operators کے JavaScript	13
44	کیا ہیں؟ Logical Operators کے JavaScript	14
44	کا استعمال کرنا if Statement میں JavaScript	15
45	کا استعمال کرنا else Statement میں JavaScript	16
47	کا استعمال کرنا else if Statement میں JavaScript	17
48	کیا کیا ہیں اور ان کا استعمال کرنا Popup Boxes میں JavaScript	18
50	کا استعمال کرنا For Loop میں JavaScript	19
52	کا استعمال کرنا while میں JavaScript	20
53	کا استعمال کرنا do while میں JavaScript	21
55	کا استعمال کرنا Switch & Case Statement میں JavaScript	22

59	Return Statement کا استعمال کرنا JavaScript میں	23
60	Break & Continue Statement کا استعمال کرنا JavaScript میں	24
63	Type Conversion کا استعمال JavaScript میں	25
74	Date and Time کا استعمال کرنا JavaScript میں	26
89	Tutorial کرنا اور ان کا استعمال کرنا JavaScript میں Cookie	27
91	DOM میں کیا ہے؟	28
94	BOM میں کیا ہے؟	29
109	Console.log() کا استعمال کرنا JavaScript میں	30
110	Timing Events کا استعمال کرنا JavaScript میں	31
112	JavaScript میں ایک گھری بنائیں گے۔	32
119	Calculator میں ایک بنائیں گے۔	33
123	Temperature Convertor میں ایک بنائیں گے۔	34

## Syntax

سب سے پہلے ہم سمجھیں گے کہ JavaScript کو لکھنے کی طرح ہیں۔  
 HTML کو لکھنے کے لئے ہم <body> یا <head> کے اندر <script> کا نیگ لگاتے ہیں اور اس کے بعد JavaScript کا کوڈ لکھتے ہیں۔

```
<script> JavaScript Code Here </script>
```

شروع کرتے ہیں یعنی درج JavaScript کا ہم پہلا کوڈ لکھتے ہیں۔ پھر ہم سمجھیں گے۔

```
<script>
document.write ("this is my first Java Script Code");
</script>
```

جبیسا کے آپ کو پہلے ہی بتاچکا ہوں کہ <script> کے نیگ کو آپ <body> یا <head> میں کہیں بھی استعمال کر سکتے ہیں تو آپ <script> کے نیگ کو کہیں بھی لکھیں کوئی فرق نہیں پڑتا۔ اب آگے بڑھتے ہیں اور کوڈ کو سمجھتے ہیں۔ ہم نے جو <script> کے نیگ کے بعد کوڈ لکھا ہے وہ "document.write" ہے اور جبیسا کے نام سے ہی ظاہر ہے اور اگر آپ سمجھ رہے ہیں کہ یہ کوئی ڈو کیو منٹ لکھے گا تو آپ بلکل صحیح ہیں۔ کیوں کہ ہم JavaScript کے کوڈ کو اپنے نیچے میں پرنٹ کرنے کے لئے اسی کوڈ کا استعمال کرتے ہیں۔  
 چونکہ JavaScript کا سارا کام بیک گرا اور نہ میں ہوتا ہے اس لئے ہمیں اس کام کو اپنے نیچے پر دکھانے کے لئے document.write کا کوڈ استعمال کرتے ہیں۔ یہ بنیادی طور پر DOM کا حصہ ہے جس کے بارے میں ہم آگے تفصیل سے پڑھیں گے۔

اب اس کوڈ کو تفصیل سے دیکھتے ہیں۔ یہ کوڈ پانچ حصوں میں ہے۔ ہم نے سب سے پہلے "document" لکھا پھر ایک " . " لگایا یہ ڈوٹ دراصل کون کینڈیشن کے لئے استعمال ہوتا ہے یعنی JavaScript کے دو کوڈ زکو آپس میں ملانے کے

لئے اور پھر اس کے بعد ہم نے "write" لکھا اور دو عدد "()" روونڈ بیکٹ (round brackets) لگانے اور اس کے بعد ہم نے ایک عدد semi-colon ":" لگایا ہے۔ اس کا کام دو ٹائمس (Statements) کو الگ الگ کرنا ہو گا ہے۔ semi-colon کو آگے تفصیل سے سمجھیں گے۔

اب شاید آپ کو یہ سب کچھ سمجھے میں مشکل ہو رہے ہوا سی لئے کوڑ دوبارہ لکھتا ہوں دیکھیں۔

```
document.write();
```

اس کوڑ کے بعد جو چیز ہمیں پرنسٹ کرنا ہوتی ہے وہ ہم بریکٹس "( )" کے اندر لکھتے ہیں۔ اور لکھنے کے لئے بھی کچھ ضروری باتیں یاد رکھیں کہ اگر ہم کوئی لفظ یا الفاظ پرنسٹ کرو ارے ہے یہ تو وہ آپ " " ڈبل یا " " سنگل کوٹس کے اندر لکھتے ہیں لیکن اگر وہ کوئی نمبر ہے تو ہم اس کو ڈائرکٹ لکھیں گے یعنی اس میں کسی بھی قسم کے کوٹس کی ضرورت نہیں ہے اس کے علاوہ اگر وہ کوئی ویریبل (variable) ہے جس کے بارے میں بعد پڑھیں گے تو وہ بھی ہم ڈائرکٹ لکھیں گے۔

اب اس کوڑ کو دیکھیں اس کوڑ اور اس کا رزلٹ دیکھیں۔

```
document.write ("this is my first Java Script Code");
```

```
document.write (555);
```

پہلے کوڑ میں ہم نے ڈبل کوٹس " " کو لگایا ہے کیونکہ وہ ایک string یعنی کے letters ہیں جبکہ دوسرے والے کوہم نے ڈائرکٹ لکھا ہے کیونکہ وہ number ہے اس لئے ہمیں اس میں کوٹس کی ضرورت نہیں۔

ان دونوں کا رزلٹ اس طرح ہو گا۔

```
this is my first Java Script Code
```

```
555
```

## Methods

JavaScript کو ویب سائچے کے اندر رائیڈ کرنے کے دو طریقے کاریں ہیں۔ ایک ہے انٹرنل (Internal) اور دوسرا ہے ایکسٹرنل (External)۔ انٹرنل کوڈ کو ہم HTML کے کسی بھی ویب سائچے کے اندر <script></script> نیگ کے اندر لکھتے ہیں۔ جیسا کہ ہم پڑھ چکے ہیں۔ مگر کوہم Script کے نیگ کو ہم HTML کے سائچے کے کسی بھی حصہ یعنی Head اور Body میں کسی بھی نیگ کے اندر لکھ سکتے ہیں۔ جبکہ ایکسٹرنل کے لئے ہم JavaScript کے لئے ایک نئی فائل بناتے ہیں جس کی ہم ایکسٹینشن ".js" رکھتے ہیں مثال کے طور میں میں JavaScript کے نام سے ایک فائل بناتا ہوں۔ اور اس کو سیوکرتا ہوں js کی ایکسٹینشن کے ساتھ دبکھیں۔

### JavaScript.js

اور اس فائل کو ہم HTML کے کسی بھی ویب سائچے کے اندر Script کا نیگ لگا کر اور Src کا آرٹیفیوٹ استعمال کر کے ویب سائچے کے اندر اس کا Path دیتے ہیں۔ جیسے کہ اگر میں نے javascript.js کی فائل بنائی اور اس فائل کو میں نے HTML کے فولڈر میں ہی رکھا ہے تو میں اس فائل کو javascript.js کا صرف نام لکھ کر Path دوں گا اور اس کو Script کے نیگ کے اندر Src کو فائل کا نام دے کر سیوکروں گا تو وہ فائل میرے ویب سائچے کے اندر رائیڈ ہو جائے گی۔

```
<script src="javascript.js"></script>
```

## Comments

دنیا کی ہر پروگرامنگ زبان کو منش کے بغیر ادھوری ہے۔ یہ وہ کوڈ ہوتے ہیں جو ویب سائچے پر نظر نہیں آتے لیکن ایک ڈو یو پر کے لئے ہوتے ہیں اور اس میں لکھی گئی بدایت کی بنابر ایک ڈو یو پر کوڈ کو آسانی سے سمجھ سکتا ہے۔ آپ لوگوں نے HTML اور CSS پہلے ہی کو منش پڑھے ہو گے اسی طرح JavaScript کے اندر بھی کو منش ہوتے ہیں۔ کو منش کا استعمال ہم اپنے کسی کوڈ کی نشاندہی یا کے

کرتے ہیں۔

JavaScript کے اندر منگل لائن کو منٹ لکھنے کے لئے آپ کو دو فارڈ سلیش کے بعد ہدایت لکھنی ہو گی اس کا طریقہ یہ ہے اور آخر میں کچھ بھی نہیں ہوتا۔

**// This is my first javascript code**

جبکہ آپ کو ایک سے زیادہ لائن کو ڈالنے کے لئے دو فارڈ سلیش کے درمیان میں دسیئرک کے درمیان ہدایت لکھنی ہوتی ہیں۔ اس کا طریقہ یہ ہے۔

**/\* This is my first javascript code  
and I am learning Javascript comment class with  
Muhammad Danish Irshad \*/**

### Data Types

JavaScript کے اندر جتنے بھی sentence Letters کے طور پر پرینٹ کرنا ہوتا ہے (String) string کہلاتے ہیں اور ان کو منگل " " اور ڈبل کوٹس " " کے اندر لکھا جاتا ہے۔ لیکن اگر ویریبل letters میں ہیں تو وہ ڈائرکٹ پرینٹ ہوں گے۔

**document.write ("this is the example of string");** مثال:-

JavaScript کے اندر جتنے بھی نمبر زہریت ہوتے ہیں ان کی ڈیٹائیاپ Integer ہوتی ہیں۔ بشرطیکہ string کے طور پر پرینٹ نہ ہو رہے ہوں۔

**مثال:-** `document.write (00923142018202);`

(۳) **Boolean:** اس کے اندر صرف Yes اور No کی کمائندہ ہوتی ہیں۔ Boolean کی ڈیٹائائمپ کا استعمال ہم آگے کلاسوں میں کریں گے۔ درج ذیل مثال میں ہم نے document.write کے فنگشن کے اندر Boolean کے فنگشن لکھا اور فنگشن سے کہا کہ اگر 9 چھوٹا ہو 10 سے تو ہماری ویڈیو صحیح ہو جائے۔ اس کا جواب true میں آئے گا۔

**مثال:-** `document.write (Boolean (10>9));`

(۴) **Array:** یہ ویریبلز کا مجموعہ ہوتا۔ ویریبلز کے بارے میں ہم آگے پڑھیں گے۔ درج ذیل مثال میں ہم نے ایک Fruits کے نام سے ویریبل بنایا جس کو ہم نے تین ویڈیویں Mango, Apple, Orange اور ان سب کو document.write کے فنگشن کے اندر پرنٹ کر دیا اور جو آپ fruits کے ویریبل کے ساتھ اسکواڑ بریکٹس دیکھ رہے ہیں ان میں ہم کو پوزیشن بتانی ہوتی ہے کہ ہمیں کون سے نمبر کی ویڈیو کا رہے۔ چونکہ ہمارے پاس تین ویڈیو ہیں اسی لئے ہم نے ہم نے نمبر 0 لکھا ہے اور کہا کہ ہمیں 0 نمبر کی ویڈیو پر پرنٹ کرنی ہے۔ نوٹ: JavaScript کے اندر پہا نمبر 0 سے شروع ہوتا ہے۔

**مثال:-** `var Fruits = ["Mango", "Apple", "Orange"];`

`document.write(Fruits[0]);`

(۵) **Object:** یہ فنگشن کا مجموعہ ہوتے ہیں۔ فنگشن کے بارے میں بھی ہم آگے پڑھیں ہیں۔ درج ذیل مثال میں ہم نے type:"Farrari", Made:2015, car کی طرح ایک Array کے نام سے ویریبل بنایا جس کو ہم نے تین ویڈیویں, color:"Red", اور ان سب کو document.write کے فنگشن کے اندر پرنٹ کر دیا اور جو آپ car کے ویریبل کے ساتھ کر لی بریکٹس دیکھ رہے ہیں صرف یہی فرق ہوتا ہے objects اور Array کے ویریبل بنانے میں لیکن اس میں آپ نے دکھا کہ میں نے type:"Farrari" لکھا ہے اس کا مطلب ہے کہ میں نے دو ویڈیویں ایک type اور اس کو بھی Farrari کی ویڈیویں نے۔ یہ والاحصہ کافی حد تک ہے یعنی کہ میں نے ایک ویریبل بنایا جس کو ایک ویڈیویں اور اس کو بھی Farrari type کی ویڈیویں۔ یہ والا حصہ کافی حد تک CSS جیسا ہے اس لئے آپ کو اس کو سمجھنے میں مشکل نہیں ہوئی چاہئے۔ لیکن object میں ہم جب پرنٹ کرتے ہیں تو ہم کو car کا

ویریبل پرنٹ کرنا ہے اپنے ویب پیچ پر لیکن یہاں ہم car کے ویریبل کے ساتھ اس ویریبل کی ولیو بھی پرنٹ کریں گے اور چونکہ ہمارے پاس تین ولیوز ہیں اس لئے ہم اس میں " . " لگا کرو وہ ولیو دیں گے جو نمیں چاہئے جیسے ہم کو بتانا ہے کہ car کی type کیا ہے تو ہم نے car کے ساتھ " . " لگا کر type لکھ دیا اور پرنٹ کرو اور یا تو جب ویب پیچ پر پرنٹ ہو گا تو ہمیں رزلٹ Farrari نظر آئے گا۔ اور object Array میں صرف اتنا فرق ہے کہ ہم کو یہاں ولیو دینا ہوتی ہے جبکہ Array میں صرف پوزیشن دینا ہوتی ہے۔ اس کے متعلق ہم آگے بھی پڑھیں گے۔

**مثال:-**

```
var car = {type:"Farrari", Made:2015, color:"Red"};
```

```
document.write(car.type);
```

## Variables

اس کلاس میں ہم ویریبل (Variable) کے بارے میں پڑھیں گے۔ variable کو سمجھنا بہت آسان ہے۔ آپ لوگوں نے الجبرا (Algebra) تو اسکول میں پڑھا ہی ہو گا۔ یعنی

$$x + y = z$$

بکل اسی طرح variables بھی ہوتے ہیں۔

**مثال:-** x کی ولیو ہے 5 اور y کی ولیو ہے 5 تو ہمیں ان کو جمع کر کے z کی ولیو معلوم کرنی ہے۔

$$x = 5;$$

$$y = 5;$$

$$z = x + y;$$

$$z = 10$$

اب اصل الجبرا میں semi-colon نہیں لگتے لیں چونکہ ہم JavaScript کھرہ ہے ہیں اسی لئے ہم نے semi-colon لگایا ہے۔ جیسا کے پہلے بتاچکا ہوں کہ Statements کو الگ الگ کرتا ہے تاکہ وہ دونوں ایک نہ ان جائیں اور کوئی Error نہ آجائے۔

مثال کے طور پر ہم نے ایک کوڈ لکھا:-

```
document.write ("hello world")
```

```
document.write ("I am writing a JavaScript Code");
```

آپ دیکھ سکتے ہیں کہ ہم نے صرف ایک کوڈ میں semi-colon لگایا ہے تو JavaScript غلط ہو گی اور یہ ایک ہی کوڈ سمجھا جائے گا جس کا مطلب ہمیں پریشانی ہونے والی ہے۔ اور بعض جگہ یہ Error بھی کھڑا کر دے گا۔ اسی لئے ہم ہر کوڈ یا statement کے بعد semi-colon لازمی لگاتے ہیں۔

اب دوبارہ variables کی طرف آتے ہیں۔ variables کی دو ناپ ہوتی ہیں ایک ہے Local variables اور دوسرا ہے Global variables ان دونوں کا فرق آگے جا کر واضح کریں گے لیکن پہلے variables دراصل ایک خالی بکس کی طرح ہوتے ہیں جس میں ترتیب سے کوئی چیز رکھی جاتی ہے اور بعد میں ضرورت کے تحت اس میں سے چیز نکال کر استعمال کی جاتی ہے۔ اسی طرح JavaScript میں آپ دیکھیں کہ کہ variables میں JavaScript کا کوڈ Save کرتے ہیں اور بعد میں استعمال بھی کرتے ہیں۔ جیسا کہ آپ نے دیکھا تھا میں نے Variable کے لئے الجبرا کی مثال دی تھی جس میں ہم نے x اور y کو 5.5 کی ویلودی تھی اور z کی ویلوان دونوں کو جمع کر کے معلوم کی تھی۔ اب x اور y کی ویلواں کچھ بھی ہو سکتی تھی اور ان دونوں کو جمع کر کے ہم z کے علاوہ a,b,c,d,e کسی کی بھی ویلواں معلوم کر سکتے ہیں۔

یقیناً ابھی سمجھنا مشکل ہو رہا ہوگا۔ لیکن جیسے جیسے ہم آگے بڑھیں گے آپ کو variable کا concept زیادہ واضح ہوتا جائے گا۔ آپ دیکھتے ہیں کہ variable کیسے کس طرح ہیں اور ان کو لکھنے کی شرائط کیا کیا ہیں۔

شراط:

(1) کوئی بھی ورپا بل انگریزی کے حرف letter سے شروع ہوگا۔

- (2) ویریبل میں نمبروں کا استعمال ہو سکتا ہے لیکن پہاڑف انگریزی کا letter یہ ہو گا۔
- (3) ویریبل میں آئیشل کریکم صرف " - " ڈیش اور " \_ " اندر اسکو ہی استعمال ہو سکتے ہیں۔
- (4) آپ ویریبل میں پسیس (Space) کا استعمال نہیں کر سکتے۔
- (5) ویریبل کا نام Reserve-Keyword میں نہیں رکھا جاسکتا۔ مثال : *if, this* if, this وغیرہ۔

اگر ان میں سے کسی بھی شرائط کو نہ مانا گیا تو Error ہو گا اور آپ کا کام نہیں ہو گا۔ اور آپ مشکل میں پڑ جائیں گے۔

Reserve-Keyword کے متعلق ہم بعد میں پڑھیں گے۔ JavaScript میں variables کو لکھنے کے دو طریقے ہیں۔ ایک طریقہ یہ ہے کہ ہم سب سے پہلے Var کا ایک Keyword لکھیں پھر اسکے بعد variable کا نام اور ایک عدد " = " یہ equal کا سائز ہے لیکن اسکا مطلب یہ equal نہیں بلکہ JavaScript میں ہم اسے Assign Operator کہتے ہیں جن کی تفصیل کے لئے ایک عابدہ کلاس ہو گی۔ اور equal کے سائز کے بعد ہم variable کو یہ لوگیں گے۔ جو کوئی نمبر، statement یہ کوئی JavaScript کا کوڈ ہو گا اس کے بعد ہم " ; " semi-colon کا سائز لگائیں گے تاکہ letter پوری ہو سکے یعنی کے variable پورا ہو گیا۔

```
var my_var_name = "write the value of variable" ;
```

اس میں آپ دیکھ سکتے ہیں کہ میں نے ایک var keyword کا استعمال کیا ہے۔ آپ دیکھ سکتے ہیں میں نے keyword کو واضح کرنے کے لئے اسکو bold اور italic کر دیا ہے جس سے آپ کو اسکی واضح پیچان ہو رہی ہو گی۔ پھر اس کے بعد میں نے variable کو ایک نام دیا جو کہ my\_var\_name کے نام سے ہے۔ میں اسکو کوئی بھی نام دے سکتا تھا۔ اس کے بعد ایک عدد equal کا سائز لگایا جو variable کو یہ Assign کرنے کے کام آتا ہے۔ پھر اس کے بعد میں نے اس کو ایک string دی یعنی کے ایک text کو یہ کے طور پر دیا کے اور چونکہ وہ letters ہیں اسی لئے وہ ڈبل کوٹس کے اندر لکھیں ہیں اور آخر میں ایک semi-colon لگایا تاکہ یہ variable پورا ہو سکے۔ اس لائن کو دوبارہ لکھنے کا مقصد صرف اتنا تھا کہ جن لوگوں کو اور پر سمجھنے میں مشکل پیش آ رہی ہو تو ان کو دوبارہ سمجھنے میں آسانی ہو۔

اس کو ایک اور انداز میں سمجھا سکتا ہوں کہ ہم نے ایک box لیا جس نام ہم نے write my\_var\_name کر دیا پھر اس کے اندر write the value of variable کو لکھ دیا اور semi-colon لگا کر بند کر دیا۔ اب جب بھی مجھے

variable کی ضرورت ہو گی میں وہ box کھولوں کا اور اس ائن کو پڑھ لوں گایا اگر کسی کو دینا ہو تو وہ دونگا۔

اب دوسرا طریقے کی طرف آتے ہیں تو وہ بھی بلکل اسی کی طرح سے صرف اس میں ہم keyword کا استعمال نہیں کرتے کیونکہ JavaScript ایک سمجھدار زبان کے اسی لئے ہم keyword کے بنا بھی variable بناتے ہیں اور ایک اور وجہ بھی ہے کہ اس کی طرف جانا ہو گا جو ہماری الگی کلاس ہو گی۔ لیکن نئے سیکھنے والوں کے لئے یہ ضروری ہے کہ وہ keyword کا استعمال کریں تاکہ انکو کسی مشکل کا سامنا نہ کرنے پڑے۔ JavaScript میں variables کا استعمال کیسے کرتے ہیں یہ دیکھ لیں۔ اب میں ہم variables کے keyword میں بتاؤں گا۔ اور الجبرا کو ہی استعمال کروں گا۔

میرے پاس ایک x ہے جس کی ولیو ہے 5 اور ایک y ہے جس کی ولیو ہے 6 اور اب مجھے ان کو جمع کر کے z کی ولیو معلوم کرنی ہے۔

```
x = 5;  
y = 6;  
z = x + y;  
z = 11
```

اب جیسا کہ آپ دیکھ سکتے ہیں ہم نے سب سے پہلے ایک x نام کا ویریبل بنایا اور اس کو 5 کی ولیو دے دی پھر اس کے بعد ایک y نام کا ویریبل بنایا اور اس کو 6 کی ولیو دے دی پھر ایک z کے نام سے ویریبل بنایا جس کے انہم نے x کو لے جمع کر دیا۔ اب ہو گا یہ کہ z کے اندر x اور y کی ولیو جمع ہو جائے گی اور ہمیں جواب z کے ویریبل کے اندر ملے گا۔ لیکن یہ ہمارے ویب پیچ کے اندر نظر نہیں آئے گا۔ کیونکہ یہ سارا کام یک گراڈ میں ہوا ہے اور اسکو ویب پیچ کے اندر show کرنے کے لئے ہمیں document.write کا استعمال کرنا ہو گا تو اس کو ہم کچھ اس طرح سے اپنے ویب پیچ کے اندر show کروائیں گے۔

```
<script>  
x = 5;  
y = 6;  
z = x + y;
```

```
document.write ("this is the answer of z = " + z);
```

```
</script>
```

اب اس کے اندر آپ دیکھ سکتے ہیں کہ document.write کے اندر ہم نے کچھ string یعنی letters پرنٹ کئے ہیں اور z کا ویریبل بھی ساتھ پرنسٹ کیا ہے۔ ایک letter ہونے کے باوجود بھی بنا سنگل یا ڈبل کوٹس کے اسی لئے لکھا گیا ہے کیونکہ وہ ایک ویریبل ہے۔ اگر ہم اس کو بھی کوٹس کے ساتھ لکھتے تو وہ ایک string ہو یا ہم جاتی اور ویریبل نہ ہتا۔ اور آپ دیکھ سکتے ہیں کہ میں نے اور ویریبل کے درمیان ایک + کا سامن بھی موجود ہے تو اس کو ہم دو ویکو جوئن کرنے کے لئے استعمال کرتے ہیں۔ جیسے ستمش کو جوئن کرنے کے لئے ہم dot (.) کا استعمال کرتے ہیں۔

اس کا رزالٹ کچھ اس طرح ہو گا۔

this is the answer of z = 11

## Functions

پہنچ فنکشن user define ہوتے ہیں جیسے کہ ہم اس کلاس میں پڑھیں گے اور پہنچ JavaScript کے اپنے ہوتے ہیں جیسے وغیرہ ان کو میں Call نہیں کرنا پڑتا جب تک کہ وہ user define فنکشن کے اندر استعمال نہ ہوں۔ فنکشن بنانے کے لئے ہمیں function کے ایک keyword کا استعمال کرنا ہو گا اور پھر اس فنکشن کا نام پھر دو راوی مذکور کیلئے اور وہ کریں بریکیلیں کا استعمال کرنا ہو گا۔ کچھ اس طرح سے دیکھیں

```
{function function-name () {}} کے فہریز
```

آپ کو سمجھانے کے لئے میں سب سے پہلے JavaScript کا استعمال کر کے سمجھاؤں گا۔ سب سے پہلے آپ ایک فنکشن بنائیں myfunction کے نام سے، نیچے دیکھیں۔

function myfunction () {}

اب ہم کر لی بریکٹ کے اندر; () کے فنگشن کا استعمال کریں گے اور اس کے اندر رائیک string میلو تحریر کریں۔ نیچے دیکھیں۔

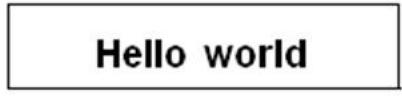
```
function myfunction () {
    document.write ("Hello world");
}
```

جی ہمارا فنگشن پورا ہوا لیکن یہ ہمارے پیچ پر ابھی بھی نظر نہیں آئے گا، حالانکہ ہم نے(); document.write() کا استعمال بھی کیا ہے لیکن اس کے باوجود بھی یہ فنگشن ہمیں نظر نہیں آئے گا۔ اسکی وجہ یہ ہے کہ کسی بھی user define فنگشن کو بنانے کے بعد اس کو پیچ کیا اندر call کرنا پڑتا ہے۔ جس کے لئے بھی ہمارے پاس کچھ طریقہ ہوتے ہیں جن میں سے ایک آپ فنگشن کا نام فنگشن کے باہر تحریر کر دیں تو وہ ویب پیچ کے اندر نظر آنے لگتا ہے۔ دیکھیں کیسے۔

```
function myfunction () {
    document.write ("Hello world");
};

myfunction ();
```

جس کے بعد اس کا رزلٹ کچھ اس طرح ہو گا۔



Hello world

امید کے آپ کو ابھی تک فنگشن کے بارے میں کافی حد تک سمجھ آئی ہو گی۔ لیکن اگر نہیں سمجھ میں آیا تو ہم ایک اور فنگشن بناتے ہیں۔ ہم اس فنگشن کو ٹین کی مدد سے اپنے پیچ کے اندر show کریں گے۔ اس فنگشن کا کام یہ ہو گا ہے HTML کے پیچ کے اندر ہمارا کچھ فلیکسٹ لکھا ہوا ہو گا اور وہ ہیں ایک ٹین ہو گا جب ہمارا یہ مر اس ٹین پر کلک کرے گا تو ہمارے پیچ کا فلیکسٹ خود بخوبی تبدیل ہو کر وہاں آپ کے کمپیوٹر کا وقت اور تاریخ تحریر کر دے گا۔ اور پھر اس کے اندر document.getElementById کے نام سے ایک کوڈ لکھا جو

HTML کے بیگ سے id سے کوئی بھی چیز اٹھایتا ہے۔ اس کام کے لئے ہم نے ایک فنگشن بنایا جس کا نام ہم نے display\_date کی مدد سے کام کر سکتے ہیں۔

display\_date کا ایک فنگشن ہے۔

document.getElementById();

HTML کے کسی بھی ٹائیک میں موجود مواد کو اس کی Attribute کی مدد سے تبدیل کر سکتے ہیں۔ اس کو لکھنے کے بعد ہمیں id کے راؤنڈ بریکٹس کے اندر id کا نام لکھنا ہے جو ہم HTML کے کسی بیگ کے اندر استعمال کریں گے۔ اس کے بارے میں یاد رکھیں کہ جیسا یہ لکھا ہوا ہے آپ کو لکھنے کے لئے ایک لفڑا بھی غلط ہوایا کوئی بھی حرفاً چھوٹا یا بڑا ہو تو سارا کوڈ بیکار ہو جائے گا اور کچھ بھی کام نہیں کرے گا۔ جو حرف چھوٹے ہیں وہ چھوٹے لکھنے کے لئے جائیں گے اور جو بڑے ہیں وہ بڑے حرف میں تحریر ہونگے۔ document.getElementById() کو تحریر کرنے کے بعد ہم ایک (dot) . کا نیس گے کیونکہ ہمیں ایک اور چیز بھی اس کے ساتھ ملانی ہے اور اسی کے ساتھ لکھنا ہے۔ جو کہ innerHTML کا ایک کوڈ ہے اس کے استعمال سے ہم HTML کے کسی بھی مواد کو تبدیل کر سکتے ہیں۔ اور ہم یہی کریں گے۔ نیچے دیکھیں کیسے لکھا ہے:-

```
function display_date( ){  
document.getElementById ("changedate") . innerHTML = Date ();  
}
```

ہم نے innerHTML کو; () Date کے ایک فنگشن کی ویلو Assign کی ہے(); Date() فنگشن کے بارے میں ہم آگے بات کریں گے۔ لیکن اس سے پہلے ہم HTML کے کا ایک <p> بیگ استعمال کریں گے۔ اور اس کو ایک id دیں گے changedate کے نام سے اور پھر اس کے بعد ہم ایک button بنا لیں گے جس میں ایک Attribute استعمال کریں گے onclick کے نام کا جس کے اندر ہم فنگشن کا نام لکھیں گے۔ نیچے دیکھیں کس طرح لکھا ہوا ہے۔

```
<p id="changedate">Welcome to this site</p>  
<button onclick="display_date ();">Click Here</button>
```

اب اس کارز لٹ دیکھیں۔

بُن پر کلک کرنے سے پہا

Welcome to this site

[Click Here](#)

بُن پر کلک کرنے کے بعد

Sat Sep 12 2015 03:12:05 GMT-0500 (Pakistan Standard Time)

[Click Here](#)

### String Method

یہاں ہم String کے methods کے بارے میں پڑھیں گے۔ دراصل حرف یا الفاظ ہوتے ہیں۔ جیسے ABCD وغیرہ۔ اس کو سمجھنے کے لئے نیچے درج مثال دیکھیں۔

```
var carName1 = "SERCI 360";
var carName2 = 'Serci 360';
document.write (carName1+"<br>"+carName2)
```

اس کے اندر ہم نے دو یہ سیل بنائے ہیں ایک carName1 اور carName2 کے نام سے اور دونوں کو ایک ہی نام دیا ہے جس کو چھوٹے اور دوسرے کو بڑے حرف میں اور ان دونوں کو document.write کے اندر پرنٹ کر دیا۔ اس کا رезولٹ دیکھیں۔

**SERCI 360**

**Serci 360**

اس میں دونوں ہنگل و یے ہی پرنسٹ ہوئے ہیں جیسے دیکھ رہے ہیں۔

اب اگلی مثال میں ہم txt.length کا استعمال کریں گے اس کے استعمال سے ہمیں کسی بھی string میلوں کے حروف کے بارے میں پتا چلتا ہے۔

```
<p id="txlh"></p>
```

یہاں ہم نے سب سے پہلے ایک p کا نیگ بنایا اور اس کے اندر id بنائی txlh کے نام سے آپ id کو کوئی بھی نام دے سکتے ہیں میں نے صرف مثال کے لئے یہ نام دیا ہے۔

```
<script>
var txt = "ABCDEFGHIJKLMNPQRSTUVWXYZ";
document.getElementById("txlh").innerHTML = txt.length;
</script>
```

اس کے بعد ہم نے script کے نیگ کے اندر اپنی اسکرپٹ لکھی۔ ہم نے پہلے txt کے نام سے ایک ویریبل بنایا جس میں ہم نے A سے لے کر پوری Z تک تمام انگریزی کے حرف تجھی لکھ لئے۔ اگلی لائن میں ہم نے document.getElementById کا استعمال کیا ہے اس کے بارے میں آگے پڑھیں گے۔ آپ اسے فناش بھی کہہ سکتے ہیں یا اونچکٹ بھی کہہ سکتے ہیں۔ ہم نے document.getElementById کے اندر اپنی p کے نیگ والی id جو کہ txlh تجھی لکھ دی اور اس کے بعد .innerHTML کا استعمال کیا اور اس کو ویودے دی txt.length کی اس سے اب جو ہمارا p نیگ ہے وہ خالی ہے لیکن اگر اس میں کچھ مواد ہوتا تو وہ تبدیل ہو جاتا .innerHTML کی وجہ سے اور نتیجہ ہی آتا جو بھی آیا ہے۔ خالی ہونے کے باوجود بھی یہ اپنا کام کرے گا اور کچھ نہ ہونے کی باوجود یہ کچھ بھی نہیں کو تبدیل کر کے txt.length کو p کے نیگ میں تبدیل کر دے گا۔ اس کا رزلٹ یہ ہو گا۔

اس اگلی مثال میں ہم چھوٹے حرف کو بڑے حرف میں تبدیل کریں گے۔ کوڈ دیکھیں۔

```
<button onclick="myFunction()">click to Upper Case</button>
<p id="upcase">Hello World!</p>
```

اس کے لئے سب سے پہلے ہم نے ایک ٹن بنایا اور اس میں event کا استعمال کیا اس کے بارے میں ہم بعد میں تفصیل سے پڑھیں گے۔ onclick کے اندر ہم نے myFunction کے نام سے جو Function بنایا ہے اس کا نام لکھا ہے اور پھر کے لئے Click to Upper Case کا نام دیا گی اور اس کو id دی کی اور اس میں کے اندر ہم نے Hello World! لکھا ہے۔

```
<script>
function myFunction() {
var text = document.getElementById("upcase").innerHTML;
document.getElementById("upcase").innerHTML = text.toUpperCase();
}
</script>
```

اس کے بعد ہم نے script کے بیگ کے اندر ایک فنكشن بنایا myFunction کے نام سے اور اس فنكشن کے اندر ایک ویرiable بنایا کی اور اس کے اندر text کے نام سے اور اس ویرiable کو ویلوا سائٹ کی document.getElementById("upcase") کی اور اس کے اندر text کی لکھدی اور اسے concadinate کر دیا۔ اس کے ساتھ اس سے اب ہماری ساری ویلوا text کے ویرiable میں محفوظ ہو گئی۔ اس کے بعد ہم نے ایک id document.getElementById("upcase") کا پھر استعمال کیا اور اس کے اندر بھی ہم نے دوبارہ id upcase کی لکھدی اور پھر سے ہم نے اسے concadinate کے ساتھ text کے ساتھ concadinate کر دیا۔ اور

.کو دیلو کے لئے ہم نے جو text کا ویریبل بنایا تھا اس کے ساتھ;() کے فناش کو .innerHTML کر کے یہ دیلو دے دی۔ concadinate

اب جب ہمارا یوزر بٹن پر کلک کرے گا تو ہمارے سارے حرف بڑے ہو جائیں گے۔

اس کا رزلٹ دیکھیں۔

کلک کے بعد میں	کلک سے پہلے
<input type="button" value="click to Upper Case"/>	<input type="button" value="click to Upper Case"/>
HELLO WORLD!	Hello World!

یہی کام ہم نے اس کوڈ میں بھی کیا ہے لیکن اس بار ہم نے(); کے toLowerCase() کے فناش کی جگہ;() کےtoUpperCase() کے فناش کا استعمال کیا ہے۔ کوڈ دیکھیں۔

```
<button onclick="myFunction()">click to Lower Case</button>
<p id="lowcase">Hello World!</p>
```

اس کوڈ میں بھی آپ دیکھ سکتے ہیں وہی سب ہے لیکن id جو پچھلے کوڈ میں upcase تھی یہاں پر lowcase ہے اور نام سے ہی ظاہر ہے یہ حرف کو چھوٹا کر دے گا۔

```
<script>
function myFunction() {
var text = document.getElementById("lowcase").innerHTML;
document.getElementById("lowcase").innerHTML = text.toLowerCase();
}
</script>
```

اس کا رزلٹ یہ ہو گا۔ اس رزلٹ میں آپ دیکھ سکتے ہیں کہ H اور W دونوں بڑے ہیں جو بعد میں لک کے بعد چھوٹے ہو گئے ہیں۔

لک کے بعد میں	لک سے پہلے
<input type="button" value="click to Lower Case"/>	<input type="button" value="click to Lower Case"/>
hello world!	Hello World!

اس اگلی مثال میں ہم لفظ کی پوزیشن نکالیں گے۔

```
<p id="cpos"></p>
```

اس میں آپ دیکھ سکتے ہیں کہ ہم نے پہلے ایک p کا ٹیگ بنایا اور اس کے اندر ایک id بنائی اور اس id کا نام ہم نے cpos دے دیا۔

```
<script>
var str = "HELLO WORLD";
document.getElementById("cpos").innerHTML = str.charAt(2);
</script>
```

اس کو ڈکے اندر ہم نے str کے نام سے ایک ویریبل بنایا اور اس کو یہ دے دی HELLO WORLD کی بڑے حرف سے پریشان مت ہوں یہاں اس سے کوئی فرق نہیں پڑتا۔ یہاں ہمیں ابھی صرف اس کی پوزیشن معلوم کرنی ہے۔ اگلی لائن میں ہم نے ایک document.getElementById کے استعمال کیا اور اس کو جو id ہم نے بنائی تھی اس کا نام دے دیا اور پھر str.charAt(2) کی اس کو str.innerHTML کے ساتھ concadinate کر دیا اور str.innerHTML کو یہ دے دی۔

میں ہم نے str.charAt() کے ساتھ concadinate کیا ہے اور جو 2 لکھا ہے اس سے ہمیں ہمارے لفظ کی پوزیشن ملے گی۔

اس کا رزلٹ یہ ہو گا۔

L

آپ دیکھ سکتے ہیں کہ ہم نے 2 لکھا تھا تو جواب L آیا جب کہ HELLO WORLD میں L تیسرا الفاظ ہے۔ لیکن JavaScript کے اندر گنتی 0 سے شروع ہوتی ہے اسی لئے L کی پوزیشن 2 نمبر پر ہے اور 0 نمبر پر H ہو گا۔

اب اگلی مثال میں ہم نے حرف کا کوڈ معلوم کریں گے اور اس کام کے لئے ہم charCodeAt() کا استعمال کریں گے۔ کوڈ دیکھیں۔

```
<p id="chcode"></p>
```

یہاں بھی ہم نے پہلے ایک p کا گلیک بنایا اور اس میں ایک id کو نام دیا chcode اور اب ہمارا سارا کام اسی id پر ہو گا۔

```
<script>
var str = "HELLO WORLD";
document.getElementById("chcode").innerHTML = str.charCodeAt(0);
</script>
```

اس کوڈ کے اندر ہم نے پہلے ہی کی طرح سارا کام کیا ہے۔ جس میں اس کا کریکٹ کوڈ مل جائے گا۔ یہاں آپ کو بتاتا چلوں کہ دنیا میں جتنے بھی لفظ یا symbols ہیں سب کا کوئی نہ کوئی کریکٹ کوڈ ہے۔ اور یہ اسی سے پہچانے جاتے ہیں۔ اس جگہ ہم نے 0 نمبر کی پوزیشن کا کوڈ مانگا ہے جو کہ حرف H ہے۔

اس کا رزلٹ یہ ہو گا۔

حرف H کا کوڈ 72 ہے۔

اب پڑھتے ہیں escape کریکٹر کے بارے میں اس کام کے لئے ہم بیشہ \ (backslash) کا استعمال کرتے ہیں۔

تفصیل	کوڈ
یہ کسی اسٹرینگ میں سنگل کوٹس کے لئے استعمال ہوتا ہے۔	\'
یہ کسی اسٹرینگ میں ڈبل کوٹس کے لئے استعمال ہوتا ہے۔	\\"
یہ کسی اسٹرینگ میں سلیش کے لئے استعمال ہوتا ہے۔	\n
یہ کسی اسٹرینگ میں نئی مائن کے لئے استعمال ہوتا ہے۔	\t
یہ بیک اپسیس کے لئے استعمال ہوتا ہے۔	\b
یہ فارم فیڈ کے لئے استعمال ہوتا ہے۔	\f

### Numbers Method

یہاں ہم Numbers کے بارے میں پڑھیں گے۔ جیسا کہ نام سے ہی ظاہر ہے یہ نمبرز ہیں۔ جیسے 54321 وغیرہ۔ اس کو مزید سمجھنے کے لئے کوڈ دیکھیں۔

```
var x = 34.00;
```

```
var y = 34;
```

```
document.write( x + "<br>" + y );
```

اس کا رезولٹ یہ ہو گا۔ اس میں decimal یا بھی بغیر decimal کے پرنت ہوتی ہے۔ لیکن اگر ہم میں decimal یا چاہئے تو اس

کا الگ طریقہ ہے۔

34

34

نمبر کے وی بیل کو ہم کسی بھی اوپر یہر کے ساتھ لکھ سکتے ہیں لیکن اگر ہم کسی نمبر کو کسی اسٹرینگ ویلو کے ساتھ جمع کریں گے تو جواب NaN آئے گا۔ کوڈ دیکھیں۔

```
document.write( 100 / "Apple" );
```

اس کا رزلٹ یہ ہو گا۔

NaN

### Math Objects

یہاں ہم Math کے انجیکٹس کے بارے میں پڑھیں گے۔ اس کو ڈیل میں ہم سب سے پہلے Math.random() کا استعمال کر رہے ہیں اس کے استعمال سے ہر بار جب آپ پیچ کو ریفرنس کریں گے آپ کو ایک نیا نمبر ملے گا۔ کوڈ دیکھیں۔

```
document.write(Math.random());
```

اس کا رزلٹ ایسا ہو گا۔ ہر بار یہ ویسا الگ ہو گی۔

0.22974724345840514

کو ہم ایک اور طریقے سے استعمال کر سکتے ہیں اس کا استعمال عام طور پر گیم کے لئے کیا جاتا ہے۔ پہاڑ کوڑ دیکھیں۔

```
document.write(Math.random() / 11);
```

اب اس کوڑ سے ہمارا نینڈ نمبر اب صرف 11 سے ہی تقسیم ہو گا اور جواب میں ہمیں ہر بار صرف 0 سے لے کر 10 تک کے نمبر زمیں گے۔ اس کا رزلٹ ایسا ہو گا۔ ہر بار یہ یلوالگ ہو گی۔

6

یہاں ہم min اور max کے بارے میں پڑھیں گے اس کو لکھنے کے لئے ہم Math.max() اور Math.min() لکھتے ہیں۔

```
document.write(Math.min(0, 150, 30, 20, -8, -200));
```

```
document.write(Math.max(0, 150, 30, 20, -8, -200));
```

یہاں ہم نے Math.max() اور Math.min() کو پانچ پانچ ویلودی ہیں ان میں سے ایک سب سے چھوٹی ویلو کو حاصل کرنے والے اور دوسرا سب سے بڑی ویلو کو حاصل کرتا ہے اور document.write() کے اندر اسے پرنٹ کرتا ہے۔

اس کا رزلٹ ایسا ہو گا۔

-200

150

یہاں ہم Math.round() کا استعمال کریں گے۔ Math.round() سب سے قریبی ویلو کو حاصل کرتا ہے۔ اس کو مزید سمجھنے

کے لئے یہ کوڈ دیکھیں۔

```
document.write(Math.round(4.4));
```

```
document.write(Math.round(4.5));
```

اس کوڈ میں ہم نے `Math.round` کے انگلیش کا استعمال کیا ہے ایک کی ویلو 4.4 ہے اور وہ سے کی ویلو 4.5 ہے۔ اور جیسا کہ سب جانتے ہیں کہ `Math` کے Rule کے مطابق اگر پونٹ یا ڈوٹ کے بعد کی ویلو 5 سے کم ہو تو پہلی ویلو ہی گنی جاتی ہے جب کہ اگر وہ ویلو 5 کے برابر یا زیادہ ہو تو پہلی ویلو میں ایک کا اضافہ کر دیا جاتا ہے۔

اس کا ریٹن ایسا ہو گا۔

4
5

math کے کچھ مزید objects یہ ہیں۔

تفصیل	کوڈ
اس سے ہم cosine معلوم کر سکتے ہیں۔	<code>cos(x)</code>
اس سے ہم کسی Angle کا tangent معلوم کر سکتے ہیں۔	<code>tan(x)</code>
اس سے ہم sine حاصل کر سکتے ہیں۔	<code>sin(x)</code>
اس سے ہم logarithm حاصل کر سکتے ہیں۔	<code>log(x)</code>
اس سے ہم سکواہر روت کا حاصل کر سکتے ہیں۔	<code>sqrt(x)</code>
اس سے ہم $x$ کے اوپر لا کا پاور حاصل کر سکتے ہیں۔	<code>pow(x,y)</code>

Events

یہاں ہم Events کے بارے میں پڑھیں گے۔ Events یہ ہے کہ کلائنٹ کے کچھ کرنے پر جو عمل آئے وہ Events ہوتے ہیں۔ مثلاً اگر کوئی ماوس کا کوئی بٹن دبائے اور یا کسی پیر اگراف کے اوپر لے جائے یا اپنے کی بورڈ سے کوئی بٹن دبائے تو اس پیر اگراف کا رنگ تبدیل ہو جائے یا اس میں کچھ الفاظ واضح ہو جائیں تو وہ Events کہاتے ہیں۔ اور وہ functions کہاتے ہیں۔ درج ذیل میں کچھ بنیادی Event واضح کر رہا ہوں جس سے آپ کو انہیں سمجھنے میں آسانی ہو گی۔

Events پر فارم کرے	Event Handling
جب پیچ لوڈ ہو	onLoad
جب کی بورڈ کی کوئی کی (key) دبائی جائے	onKeydown
جب کی بورڈ کی کوئی کی (key) مسلسل دبائے رکھی جائے	onKeypress
جب کی بورڈ کی کوئی کی (key) دبائی رکھی ہو وہ اوپر واپس آئے	onkeyup
جب ماوس سے click کیا جائے	onClick
جب ماوس سے double click کیا جائے	onDbClick
جب ماوس کا بٹن دبایا جائے	onmousedown
جب ماوس کا دبایا ہوا بٹن انٹھ جائے	onMouseup
جب ماوس کسی object پر لایا جائے	onMouseenter
جب ماوس کسی object سے ہٹایا جائے	onMouseleave
جب ماوس کو کسی object کے اوپر لایا یا کسی object کے اوپر لاایا جائے	onMouseover
جب ماوس کو کسی object کے اوپر حرکت دی جائے یا گھماایا جائے	onMousemove
جب ماوس کسی object سے ہٹایا جائے	onMouseout
جب یوزر کوئی چیز مثلاً Text وغیرہ سائیکٹ کرے	onSelect
جب یوزر submit کے بٹن پر کلک کرے یا فارم کرے	onSubmit

اب ایک کوڈ دیکھیں

```
<h2 onclick="this.style.color= 'blue';">
```

**Click Text to Turn Blue**

```
</h2>
```

اس کوڈ میں ہم نے ایک Attribute کا استعمال کیا ہے۔ یہ HTML کا جواہ اسکرپٹ کے لئے ہی بنائی ہے اور ہمارے events میں سے بھی ایک ہے۔ اس کے اندر ہم نے this.style.color کا استعمال کیا ہے اور اس کو ویلودے دی blue کا استعمال ہم تکرتے ہیں جب ہمیں JavaScript کے اندر کوئی CSS کا اشائیں دینا ہو۔ اس کوڈ کے تین حصے ہیں۔ پہلا ہے اس سے ہم اس کوڈ کو بتاتے ہیں کہ جو کام کرنا ہے وہ یعنی پر کرنا ہے۔ دوسرا style اس سے ہم کوئی بھی CSS کا اشائیں لگانے کے لئے ہیں مثلاً style.color، style.position، style.size، style.fontSize اور آخر میں ہم نے جو color لکھا ہے اس سے ہمیں پتا چل جاتا ہے کہ یہ اشائیں کر کے لئے ہے اور اس اشائیں کو ہم نے ویلودے دی blue کی جس سے اب اس کوڈ کی وجہ سے جب بھی کوئی بھی یوزر تحریر پر کلک کرے گا تو وہ نیلے رنگ کی ہو جائے گی۔ یہ خصوصیت کے لئے تھا اگر مشکل ہو رہی ہے تو اگلی مثال دیکھیں۔

```
<div onmouseover="this.style.color= 'blue';
```

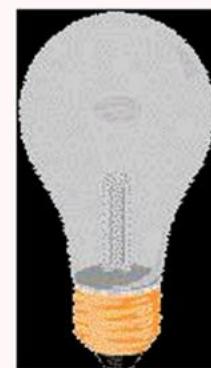
```
this.style.fontSize='20px;'">
```

**Click Text to Turn Blue and increase font size**

```
</div>
```

اس کوڈ میں ہم نے event کا استعمال کیا ہے۔ اس کوڈ سے جب کوئی یوزر تحریر پر کلک کرے گا تو اس کا سائز 20px اور کلر نیلا ہو جائے گا۔

اب یہی کام ہم دو تصویروں کے ساتھ کرتے ہیں اس کام کے لئے ہم مندرجہ ذیل دو تصویروں کا استعمال کریں گے۔



```

```

اس کوڈ میں `this.src` سے تصویر کا سورس تبدیل ہو جائے گا۔ جب کوئی یوزر تصویر پر اپنا ماوس لے کر جائے گا تو تصویر تبدیل ہو جائے گی اور بند بلب کی جگہ روشن بلب کی تصویر آجائے گی اور ایسا لگے گا جیسے ماوس لے کر جانے سے بلب جل گیا ہو اور جب ماوس دوبارہ ہٹایا جائے گا تو پھر پہلے والی تصویر دوبارہ آجائے گی اور بلب پھر بند ہو جائے گا۔

```
<h2 onmouseover="this.innerText='mouse out to change text again';"
onmouseout="this.innerText='Thank you';">
mouse over me to change text
</h2>
```

اس مثال میں ہم نے `this.innerText` کا استعمال کیا ہے اس سے ہماری تحریر تبدیل ہو جائے گی۔

## Assignment Operator

یہ آسانمنٹ اور پیٹر ہے۔

### Assign

=	Assign
---	--------

یہ کسی بھی وریبل کو ویلودینے کے کام آتا ہے یہ equal نہیں اس کام کے لئے ہم اسے دوبار استعمال کرتے ہیں۔ مثال کے طور پر ہم ایک وریبل کو ایک ویلودینا چاہتے ہیں تو اس کو ویلودینے کے لئے ہم " = " کے ساتھ کاستعمال کریں گے تاکہ اس وریبل کو ویلود سکے۔ یہاں ہم نے ایک وریبل بنایا myvar\_1 کے نام سے اور اس کو ایک String ویلود دی۔

**Var myvar\_1 = "the data of variable is string";**

## Arithmetic Operators

یہ آرٹھرایٹک اور پیٹر ہے۔

### Addition

+	Add
---	-----

یہ دو وریبلو یاد نمبرز کو جمع کرنے کے لئے استعمال ہوتا ہے۔ مثال کے طور پر ہم نے دو وریبل بنائے۔ ایک x کے نام سے جس کو ہم نے 5 کی ویلود دی اور ایک y کے نام سے جس کو ہم نے 5 کی ویلود دی اور دونوں کو ہم نے document.write(); کے نتائج کے اندر جمع کا اور پیٹر لگا کر جمع کر دیا۔

```
var x = 5;
var y = 5;
document.write(x + y);
```

اس کا ریز لٹ یہ ہو گا۔

10

یہاں + کا سائز ان دو نوں کو ایک ساتھ ملائ کر نہیں لکھنے کے لئے استعمال نہیں ہو رہا کیونکہ یہ دو نوں ویریبل ہیں اور دونوں کو جمع ہونا ہے۔  
اگر یہ سٹرنگ ہوتے اور سنگل یا ڈبل کوٹس کے اندر لکھنے ہوتے تو آپ دونوں x اور y اگل الگ لکھنے ہوئے دیکھتے۔

### Subtract

-

**Subtract**

یہ دو ویریبل یا دو نمبر زکو منقی کرنے کے لئے استعمال ہوتا ہے۔ مثال کے طور پر ہم نے دو ویریبل بنائے۔ ایک x کے نام سے جس کو ہم نے 5 کی ولیودے دی اور ایک y کے نام سے جس کو ہم نے 15 کی ولیودے دی اور دونوں کو ہم نے document.write(); کے تنگشن کے اندر ماٹنس کا اور پر یہڑا لگا کر رکھتا کر دیا۔

```
var x = 5;
var y = 15;
document.write(x - y);
```

اس کا ریز لٹ یہ ہو گا۔

5

**Multiplication**

\*

**Staric / Multiplay**

یہ اسیئر کا سائنس ہے۔ یہ دوویری بدلر یا دو نمبر کو ضرب (x) کرنے کے لئے استعمال ہوتا ہے۔ مثال کے طور پر ہم نے دوویری بدل بنائے۔ ایک x کے نام سے جس کو ہم نے 5 کی ویلودے دی اور ایک y کے نام سے جس کو ہم نے 5 کی ویلودے دی اور دونوں کو ہم نے **function** کے اندر اسیئر کا سائنس لگا کر ضرب کر دیا۔

```
var x = 5;
var y = 5;
document.write(x * y);
```

اس کا ریز لٹ یہ ہو گا۔

25

**Division**

/

**Slash / Divide**

یہ سدیش کا سائنس ہے جو کہ تقسیم (÷) کے لئے استعمال ہوتا ہے۔ یہ دوویری بدلر یا دو نمبروں کو تقسیم کرنے کے لئے استعمال ہوتا ہے۔ مثال کے طور پر ہم نے دوویری بدل بنائے۔ ایک x کے نام سے جس کو ہم نے 15 کی ویلودے دی اور ایک y کے نام سے جس کو ہم نے 3 کی

ویلودے دی اور دونوں کو ہم نے `document.write();` کے نتائج کے اندر سلیش کا سائز لگا کر تقسیم کر دیا۔

```
var x = 15;
```

```
var y = 3;
```

```
document.write(x / y);
```

اس کا رزولٹ یہ ہو گا۔

5

## Modulas

%	<b>Percentage / Modulas</b>
---	-----------------------------

اس کو سمجھنا جو گیا یہ مودولوس یا پر سنتیج کا سائز کہا جاتا ہے۔ یہ بھی Divide کی طرح کام کرتا ہے لیکن ہونے کے بعد جو رقم فوج جاتی ہے وہ رقم یہ دیتا ہے۔ جیسے ہم نے پہلے Divide کیا تو 3 کو پانچ مرتبہ جمع کرنے سے 15 بنتا ہے یعنی 15 کو 3 سے ضرب کرنے پر جواب 5 ہوتا ہے تو وہ Divide کا جواب تھا۔ لیکن مودولوس میں ایسا نہیں ہوتا یہاں 3 اگر 5 سے ضرب ہو جاتا ہے تو جواب 0 آئے گا۔ لیکن اگر 3 کو 17 سے ضرب کریں تو 15 تک ضرب کرنے کے بعد جب ضرب نہیں ہو سکے گا تو جو فوج جائے گا وہ جواب ہو گا یعنی کے 2 جواب ہو گا۔ اس کو سمجھنے کے لئے دو مثالیں دیکھیں۔

مثال 1 میں ہم نے دو ویریبل بنائے۔ ایک x کے نام سے جس کو ہم نے 15 کی ویلودے دی اور ایک y کے نام سے جس کو ہم نے 3 کی ویلودے دی اور دونوں کو ہم نے `document.write();` کے نتائج کے اندر جمع کا اور پریٹر لگا کر جمع کر دیا۔

```
var x = 15;
```

```
var y = 3;
```

```
document.write(x % y);
```

اس کا رزالٹ یہ ہوگا۔

0

مثال 2 میں ہم نے دو ویریبل بنائے۔ ایک x کے نام سے جس کو ہم نے 17 کی ویلودے دی اور ایک y کے نام سے جس کو ہم نے 3 کی ویلودے دی اور دونوں کو ہم نے `document.write()` کے ٹشن کے اندر جمع کا اور پھر لگا کر جمع کر دیا۔

```
var x = 17;
var y = 3;
document.write(x % y);
```

اس کا رزالٹ یہ ہوگا۔

2

یہاں آپ دیکھ سکتے ہیں 3 جب تک ضرب ہو ستا تھا ہو گیا یعنی کے 15 تک لیکن چونکہ 3 کہ پیہاڑے میں 17 نہیں ہوتا اس لئے وہ وہ بیس رک گیا اور جواب ہمارے پاس 2 آگیا۔ اگر آپ کو سمجھنے نے مشکل لگتا تو بار بار پڑھیں اور پر کشیں کریں۔ یقیناً جلد سمجھا جائے گا۔

### Incresement ++

++

Incresement ++

یہ سائن دیکھ کر آپ کو لگ رہا ہوگا کہ یہ دوبار جمع کرے گا لیکن ایسا نہیں ہے۔ یہ دوبار جمع نہیں کرتا بلکہ یہ کسی بھی نمبر کے اندر 1 کا اضافہ

کر دیتا ہے۔ مثال کے طور پر ہم نے ایک ویریبل بنایا۔ x کے نام سے جس کو ہم نے 5 کی ولیودے دی اور اس کو لکھنے سے پہلے ++ کا سائنس لگا دیا۔ اس کو ہم نے document.write(); فناشن کے اندر پرنسٹ کر دیا۔ یاد رکھیں ++ کا سائنس آپ کو پہلے لگانا ہے بعد میں ورنہ یلوئیں دے گا۔

```
var x = 5;
document.write(++x);
```

اس کا رزلٹ یہ ہو گا۔

6

### Decresement --

#### **Decresement --**

++ والے سائنس کی طرح یہ بھی دوبارہ نہیں کرتا بلکہ یہ کسی بھی نمبر کے اندر 1 کی کمی کر دیتا ہے۔ مثال کے طور پر ہم نے ایک ویریبل بنایا۔ x کے نام سے جس کو ہم نے 5 کی ولیودے دی اور اس کو لکھنے سے پہلے -- کا سائنس لگا دیا۔ اس کو ہم نے فناشن کے اندر پرنسٹ کر دیا۔ دوبارہ بتارہ ہوں کہ یاد رکھیں -- کا سائنس آپ کو پہلے لگانا ہے بعد میں ورنہ جو یلو آپ چاہتے ہیں وہ آپ کو نہیں ملے گی۔

```
var x = 5;
document.write(--x);
```

اس کا رزلٹ یہ ہو گا۔

4

## Comparison Operators

یہ کمپیوٹر اور پریمیرز میں۔

### Equal

==	Equal
----	-------

ہم جیسا کہ پہلے پڑھ چکے ہیں ڈبل equal کا مطلب ہے تو اس کا استعمال ہم تکرتے ہیں جب ہمیں یہ دیکھنا ہو کہ ہمارے پاس جو دو ویلوز ہیں وہ ایک برابر ہیں یا نہیں۔ مثال کے لئے ہم دو ویلوز بناتے ہیں ایک x کے نام سے اور اس کو 5 کی ولیو دے دیتے ہیں اور دوسرا لاء کے نام سے اور اس کو بھی ہم 5 کی ولیو دے دیتے ہیں اور ایک if کے فنگشن کا استعمال کر کے دیکھتے ہیں کہ یہ دونوں برابر ہیں یا نہیں۔ اگر یہ دونوں برابر ہونگے تو ایک میسح آئے گا کہ یہ دونوں برابر ہیں ورنہ کوئی میسح نہیں آئے گا۔ ہم if کے بارے میں آگے تفصیل سے پڑھیں گے۔

```

var x = 5;
var y = 5;
if ( x == y ) {
    document.write("The two variables are equal");
}

```

4

اس کا رزالٹ یہ ہو گا۔

The two variables are equal
-----------------------------

**Equal to Type****====****Equal to Type**

اس میں آپ دیکھ رہے ہیں کہ اس میں تین equal کے سائنس ہیں اس کو ہمتب استعمال کرتے ہیں جب ہمیں معلوم کرنا ہو کے اگر ہمارے پاس جوڈیٹا ہے اس کی ویلو اور ڈیٹا میپ دونوں برابر ہیں یعنی اس کو سمجھنے کے لئے ہم دوویریبل بناتے ہیں۔ ایک x کے نام سے جس کو ہم نے 5 کی ویلو دیتے ہیں اور ایک y کے نام سے جس کو ہم نے 5 کی ویلو دیتے ہیں اور پھر ہم ایک فنکشن استعمال کرتے ہیں if کا اور دیکھتے ہیں کہ آیا اگر دونوں کی ویلو اور ڈیٹا میپ ایک ہیں تو ہمیں ایک مسج اپنے پیچ پر دکھاؤ۔

**var x = 5;****var y = 5;****if (x === y) {**

```
document.write("both variables have same DataType and Value");
}
```

اس کا رزلٹ یہ ہو گا۔

**both variables have same DataType and Value****Not****!****Not**

جب یہ کہا میں سے ہی ظاہر ہے یہ Not یعنی نہیں کے لئے استعمال ہوتا ہے۔ مثال کے طور پر اگر ہم چاہیں کہ ہمارے پاس دوویریبلوں ہوں اور اگر دونوں کی ویلو ایک نہ ہو تو ہمیں ایک پیغام دیب پیچ میں نظر آئے تو اس کا Not کا استعمال کرتے ہیں۔ اس کو سمجھنے

کے لئے ہم دو یہ بدلہ بناتے ہیں۔ ایک کو ہم `x` کا نام دیتے ہیں جس کو ہم نے 5 کی ویلودیتے ہیں اور دوسرے کو ہم `y` کا نام دیتے ہیں جس کو ہم 6 کی ویلودیتے ہیں اور اس کے بعد ہم ایک `if` کا فنگشن بناتے ہیں اس فنگشن کے اندر ہم چاہتے ہیں کہ اگر ہمارے دو فوں دو یہ بدلہ کی ویلودیتے ہیں ایک پیغام نظر آئے۔ اس کام کے لئے ہم `==` کے ساتھ کے ساتھ کے بھی استعمال کریں گے۔

```
var x = 5;
var y = 6;
if (x !== y) {
    document.write("variable x and y has not same value");
}
```

اس کا ریز لٹ یہ ہو گا۔

### variable x and y has not same value

اسی طرح ہم چاہتے ہیں کہ اگر دو فوں کی ناٹپ بھی ایک نہ ہو تو بھی ہمیں ایک پیغام نظر آئے تو ہم اس کام کے لئے تین بار `equal` کا ساتھ لگائیں گے اور `y` کو ہم `string` میں دیں گے۔

```
var x = 5;
var y = "Hello";
if (x !== y) {
    document.write("variable x and y has not same Data type");
}
```

اس کا ریز لٹ یہ ہو گا۔

variable x and y has not same Data type

Greater Than

&lt;

Greater Than

یہ HTML کے بیگ کو شارٹ کرنے کا پہلا سائنس ہوتا ہے۔ اس کا استعمال ہم کسی بھی دو ویلوز میں بڑے یا چھوٹے کی تفریق کے لئے استعمال ہوتا ہے۔ مثال سے صحیح ہم نے دو ویریبل بنائے۔ ایک کو x کا نام دیا اور اس کو ہم نے 5 کی ولیودے دی اور ایک کو y کا نام دیا اور اس کو ہم نے 6 کی ولیودے دی اور اس کے بعد ہم نے ایک if کا فنکشن بنایا۔ اس فنکشن کے اندر ہم چاہتے ہیں کہ اگر ہمارے دونوں ویریبلوں میں سے y کی ولیو بڑی ہو تو ہمیں ایک پیغام نظر آئے۔

```
var x = 5;
```

```
var y = 6;
```

```
if(x < y) {
```

```
document.write("value of the y is greater than x");
```

```
}
```

اس کا ریز لٹ یہ ہو گا۔

value of the y is greater than x

Less Than

&gt;

Less Than

یہ HTML کے بیگ کو ختم کرنے کا دوسرا سائنس ہوتا ہے۔ اس کا استعمال بھی ہم کسی دو ویلوز میں بڑے یا چھوٹے کی تفریق کے لئے

کرتے ہیں۔ مثال سے سمجھئے ہم نے دو ویریبل بنائے۔ ایک کو x کا نام دیا اور اس کو ہم نے 5 کی ولیودے دی اور ایک کو y کا نام دیا اور اس کو ہم نے 4 کی ولیودے دی اور اس کے بعد ہم نے ایک if فنکشن بنایا۔ اس فنکشن کے اندر ہم چاہتے ہیں کہ اگر ہمارے دونوں ویریبلوں میں سے y کی ولیوچھوٹی ہو تو ہمیں ایک پیغام نظر آئے۔

```
var x = 5;
var y = 4;
if (x > y) {
document.write("value of the y is Less then x");
}
```

اس کا ریز لٹ یہ ہو گا۔

value of the y is Less then x

### Greater Then or Equal

<=

Greater then or Equal

جبیسا کہاں سے ہی ظاہر ہے اگر ایک ویریبل کی ولیو درمیں ویریبل سے زیادہ ہو یا پھر کم از کم اس کے برابر ہو تو ریز لٹ ہمیں دے۔ اس کام کے لئے بھی ہم if کے فنکشن کا استعمال کریں گے۔

مثال نمبر 1:-

```
var x = 5;
var y = 5;
```

```

if (x <= y) {

document.write("Value of the variable y is Greater then equal to x");

}

```

اس کا ریز لٹ یہ ہو گا۔

**Value of the variable y is Less then or equal to x**

مثال نمبر 2:-

```

var x = 5;

var y = 8;

if (x <= y) {

document.write("Value of the variable y is Greater then equal to x");

}

```

اس کا ریز لٹ یہ ہو گا۔

**Value of the variable y is Greater then or equal to x**

**Less Then or Equal**

>=	Less Then or Equal
----	--------------------

جیسا کہ میں سے ہی ظاہر ہے اگر ایک ویریبل کی دلیل و میرے ویریبل سے کم ہو یا پھر کم از کم اس کے برابر ہو تو ریز لٹ ہمیں دے اور اس

کام کے لئے بھی ہم if کے فکشن ہی کا استعمال کریں گے۔

مثال نمبر 1:-

```
var x = 5;
var y = 5;
if (x >= y) {
document.write("Value of the variable y is Less than or equal to x");
}
```

اس کا ریز لٹ یہ ہو گا۔

**Value of the variable y is Less than or equal to x**

مثال نمبر 2:-

```
var x = 5;
var y = 8;
if (x >= y) {
document.write("Value of the variable y is Less than or equal to x");
}
```

اس کا ریز لٹ یہ ہو گا۔

**Value of the variable y is Less than or equal to x**

## Logical Operators

یہ وجہ میں اور پریمیرز ہیں۔

## Addition

**+ Add**

یہ دو ویریبلز یا دو نمبرز کو جمع کرنے کے لئے استعمال ہوتا ہے۔ مثال کے طور پر ہم نے دو ویریبل بنائے۔ ایک x کے نام سے جس کو ہم نے 5 کی ولیودے دی اور ایک y کے نام سے جس کو ہم نے 5 کی ولیودے دی اور دونوں کو ہم نے document.write(); کے ف麟شن کے اندر جمع کا اور پریمیر لگا کر جمع کر دیا۔

```
var x = 5;  
var y = 5;  
document.write(x + y);
```

اس کا رز لٹ یہ ہو گا۔

10

if  
≡

کام مطلب ہے اگر مثال کے لئے انگریزی کا جملہ "if you have a bike" اگر آپ کے پاس بائگ ہے۔ اس جملے میں کمی نظر if آتی ہے جیسے اگر آپ کے پاس بائگ ہے تو آپ کیا کریں گے؟ تو اب ہمیں یہ بتانا ہے کہ اگر آپ کے پاس بائگ ہے تو آپ کیا کریں۔ بھی اسی کام کے لئے استعمال ہوتا ہے۔ if کے فنگشن کو لکھنے کے لئے ہم function کی طرح if

کے بعد دوڑا ڈنڈ بریکلیٹس اور دو کری بریکلیٹس کے اندر ہم کنڈیشن لکھتے ہیں اور کری بریکلیٹس کے اندر ہم کوڈ لکھتے ہیں جو کام ہم کو کروانا ہوتا ہے۔

```
if () {}
```

اس کو سمجھنے لئے ہم ایک کوڈ لکھتے ہیں۔ سب سے پہلے دو یونیٹ بنائیں a اور b کے نام سے اور ان کو مختلف ویلوز دیں۔ پھر if کے تناشن کے اندر ان کو لکھیں کہ آیا اگر a کی ویلوز یاد ہوتی تو ہمیں ایک منیج دیکھا دیں۔ یعنی اگر a کی ویلوز b سے بڑی ہوگی تو ہمیں رزلٹ ملے گا اور نہ ہمیں کچھ نظر نہیں آئے گا۔

```
var a = 12;
var b = 15;
if ( a < b ) {
document.write ("if you have a bike go to bazaar");
}
```

اس کا رزلٹ یہ ہو گا۔

```
if you have a bike go to bazaar
```

else

else کو سمجھنا بھی آسان ہے۔ else کا مطلب ہے مگر یعنی if اگر اور else مگر اس کو سمجھنے کے لئے ایک انگریزی کا جملہ دیکھیں۔

"if you have a bike go to bazaar else"

اس میں ہم نے یہ تو بتا دیا کہ "اگر آپ کے پاس بانگ ہے تو آپ اپنے بازار جائیں ورنہ" اب ہمیں اس کو یہ بتاتا ہے کہ ورنہ وہ کیا کرے اگر اس کے پاس بانگ نہ ہو تو وہ پھر اس صورت میں کیا کرے۔ تو ہم else کا استعمال اس لئے کرتے ہیں کہ اگر ایسا نہ ہو تو اور کیا کرنا چاہئے۔ else کو لکھنے کا طریقہ بھی بہت آسان ہے آپ else لکھیں اور وہ کریں بلکہ دیں کیونکہ اس میں آپ کو کندھیں کی ضرورت نہیں۔ ایک سادہ سی مثال دیکھیں اور پروالی ہی مثال کو آگے لے کر چلتے ہیں۔

```
var a = 12;
var b = 10;
if ( a < b ) {
    document.write ("if you have a bike go to bazaar");
}
else {
    document.write ("if you have a bike go to bazaar else take a bus and go
to bazaar");
}
```

اس کا رезولٹ یہ ہو گا۔

**if you have a bike go to bazaar else take  
a bus and go to bazaar**

آپ یہاں غور کریں کہ میں نے if کے فنگشن میں greater then کا سائز تبدیل نہیں کیا صرف b کی ویلوکم کر دی ہے۔ جس کا مطلب ہے کہ اب b کی ویلوکم ہو گئی ہے۔ جب ہم پہلے کام کرتے تھے تو ہمارے پاس اس غلطی کے بعد کوئی جواب نہیں آتا تھا لیکن اب چونکہ ہم نے else کا آپشن بھی استعمال کیا ہے اسی لئے اب ہمارے پاس ایک اور جواب بھی موجود ہے۔ جو اپنے نظر آ رہا ہے۔ یعنی اگر a کی ویلوکم سے بڑی ہو گی تو ہمیں پہاڑا والارزولٹ نظر آئے گا اور ندوسرا والارزولٹ نظر آئے گا۔

**else if**

یہ بھی if کی ہی طرح ہے اس کا کام بھی یہی ہے لیکن ہم اس کو جب استعمال کرتے ہیں جب ہمیں کسی کام کے متعلق مختلف آپشنز رکھنا ہوں۔ else if کو مثل کرنے ہم انگریزی کے اسی جملے کو پھر دہراتے ہیں۔

"if you have a bike go to bazaar else take a bus and go to bazaar or give me back my money"

اب آپ دیکھ سکتے ہیں کہ اس بار جملے میں ہم نے کہا کہ اگر آپ کے پاس بانک ہے تو آپ بازار جائیں ورنہ آپ بس پکڑیں اور بازار جائیں یا پھر میرے پیسے واپس کر دیں۔

```

var a = 12;
var b = '10';
if ( a < b ) {
    document.write ("if you have a bike go to bazaar");
}
else if (a > b) {
    document.write ("if you have a bike go to bazaar else take a bus or give
me back my money");
}
else {document.write ("if you have a bike go to bazaar else take a bus");
}

```

اس کا رезولٹ یہ ہو گا۔

**if you have a bike go to bazaar else take a bus and go**  
**to bazaar or give me back my money**

## Popup Boxes

تین مختلف اقسام کے ہوتے ہیں۔ جو درج ذیل ہیں۔ یہ بنیادی طور پر Browser Object Model کا حصہ ہیں۔ جن کے بارے میں ہم آگے تفصیل سے پڑھیں گے۔

### 1) Alert Box

سب سے پہلا Popup Box ہے جس کا استعمال ہم اپنے ویپ یقیں کے اندر user خوش آمدید کرنے کے لئے کسی بھی قسم کی وارنگ دینے کے لئے کرتے ہیں۔ لیکن اس کے استعمال کو زیادہ تر لوگ پسند نہیں کرتے۔ اور کیوں جب آپ کو خود ہی پتا چل جائے گی جب آپ اس کا استعمال کریں گے۔ پھر بھی میں آپ کو سکھا رہا ہوں۔ دیکھیں کہ alert box کو کس طرح لکھتے ہیں۔

```
alert ("Hello, How are you?");
```

یہ ایک نگاش ہے۔ ہم اس کو ویریبل کے اندر بھی استعمال کر سکتے ہیں۔ اور بعد از ضرورت استعمال میں لاسکتے ہیں۔ اس کا زائد سچھا اس طرح کا ہو گا۔



### 2) Prompt Box

ہمارا Popup Box ہے جس کا استعمال ہم اپنے ویپ یقیں کے اندر user سے کسی بھی قسم کا ڈیٹا حاصل کرنے کے لئے کرتے ہیں۔ اس کے دو پیرامیٹرز ہوتے ہیں۔ ایک جو اپر ڈسکریپشن کے طور پر ہوتا ہے اور دوسرا جو کہ user کے ڈیٹا ان پر فیلڈ کے اندر ہوتا ہے۔ دیکھیں کہ prompt box کو کس طرح لکھتے ہیں۔

```
prompt ("Please enter your name" , "Danish Irshad");
```

اب جیسے ہی اس **user** کے بڑھن پر کلک کرے گا وہ ہمارے ویب سائچے کے اندر آجائے گا۔ ہم اس کو ویریبل کے اندر بھی استعمال کر سکتے ہیں۔ دیکھیں کیسے:-

```
var name = prompt ("Please enter your name" , "Danish");
document.write ("Hello, " + name + " Welcome to our site");
```

اب اس کا رزلٹ کچھ اس طرح ہو گا۔

**Hello, Danish Welcome to our site**

### 3) Confirm Box

اس کا استعمال ہم اپنے ویب سائچے میں **user** کے کسی چیز کے متعلق ہاں یا نامیں جواب مانگتے ہیں۔ مثال کے طور پر اگر ہم اپنے **user** کو کہیں کہ اگر آپ ہماری شرائط کو قبول کرتے ہیں تو آپ ہماری ویب سائٹ کے اندر داخل ہو سکتے ہیں ورنہ نہیں۔ اس کو اس طرح لکھتے ہیں دیکھیں۔

```
confirm ("Do you agree to our Terms & Services");
```

لیکن ابھی بھی اگر کوئی یوزر **No** پر کلک کرتا ہے تو وہ بھی ہمارے سائچے کے اندر روزت کر سکتا ہے کیونکہ ہم نے اس کو کوئی تباہی راستہ کہیں جانے کے لئے نہیں دیا۔ اسی لئے ہمیں اس کے اندر **if** اور **else** کے فکشن کا استعمال کرنا ہو گا۔

```
var x=confirm("Press a button!");
```

```
if(x == true) {
```

```

document.write ("You Pressed Ok!");

} else {

document.write ("You Pressed Cancel!");

}

```

ہم نے یہاں سب سے پہلے ایک x کے نام سے ویریبل بنایا جس کو ہم نے دیلوں(); confirm() کی دے دی۔ پھر ہم نے ایک if کا فنگشن بنایا جس میں ہم نے if کے فنگشن کو کہا کہ اگر ویریبل x کی دیلوں جو کہ confirm box ہم نے دی ہے۔ اگر برابر ہو تو صحیح دیں اور document.write کے ذریعے سے You Pressed ok! پر پہنچ کر دیں۔ جس سے اگر کوئی یوزر ok پر کلک کرے گا تو دو یہ صحیح ہو جائے گی۔ لیکن اگر کوئی یوزر cancel پر کلک کرتا ہے تو ہم نے ابھی تک نہیں بتایا کہ جاوا اسکرپٹ کو کیا کرنا ہے۔ ہم نے اس کے لئے else کے فنگشن کا استعمال کیا اور جیسا کہ آپ لوگ جانتے ہیں کہ else کا مطلب ہی مگر ہے۔ یعنی کہ ہمیں وضاحت کرنے کی ضرورت نہیں۔ اگر کوئی یوزر cancel پر کلک کرتا ہے تو پر else کے فنگشن کی مدد سے فنگشن کے document.write کے You Pressed Cancel! پر پہنچ ہو گا۔

اس کا رزلٹ کچھ اس طرح ہو گا۔



### for loop

لوب تمیں مختلف اقسام کے ہوتے ہیں۔ for loop ہماری لوب کی کاس کا سب سے پہلا لوب ہے۔ یہ لوب سب سے زیادہ مشہور اور سب سے زیادہ استعمال ہونے والا لوب ہے۔ تینوں لوبس میں بہت زیادہ فرق نہیں ہے۔ لیکن فور لوب کے اندر ہم کسی بھی ویریبل کو اس کے اندر ہی استعمال کرتے ہیں۔ فور لوب اس طرح لکھا جاتا ہے۔

for () {}

اس کو تفصیل سے سمجھنے کے لئے آپ نیچے درج کوڈ کو دیکھیں پھر میں آپ کو اس کوڈ کو سمجھاتا ہوں۔

```
for (x=1; x<10; x++) {  
    document.write(x);  
}
```

ہم نے سب سے پہلا ایک فور لوپ بنایا جس کے اندر ہم نے ایک ویریبل بنایا x کے نام سے اور اس کو ویلودے دی 1 کی پھر ہم نے x کے ویریبل کو اس میں ڈیفائل کیا جس کو ہم نے greater than کا سائیں لگا کر ایک ویلودے دی جس کو ہم نے 10 کی ویلودے دی جس سے اس کوڈ کو ہم نے بتایا کہ اگر x کی ویلودے 10 سے چھوٹی ہو تو اس میں ایک کا اضافہ کرو۔ اب ہمیں کیسے پتا چلے گا کہ یہ اضافہ کہاں ہو رہا ہے اور کس چیز میں ہو رہا ہے تو اس کام کے لئے ہم نے document.write کا استعمال کیا جس میں ہم نے x کو لکھ دیا۔ اب یہ لوپ کام شروع کر چکا ہے جو لگاتار چلتے رہیں گے۔ جب تک کہ کنڈیشن غلط نہیں ہو جاتی۔ یعنی کہ جیسے ہی ہمارا 1 نمبر 10 کے برابر ہو جائے گا کنڈیشن غلط ہو جائے گی اور رزلٹ پر نہ ہونا بند ہو جائے گا اور لوپ نگ بند ہو جائے گی۔ اب جب آپ رزلٹ دیکھیں گے تو آپ کو 1 سے لے کر 9 تک نمبر زیلیں گے۔

اس کا رسلٹ یہ ہے۔

123456789

ہم کے بجائے Dicresement کا سائیں بھی استعمال کر سکتے ہیں۔ کوڈ دیکھیں۔

```
for (x=20; x>10; x--) {  
    document.write(x);
```

}

ہم نے ایک ویریبل بنایا x کے نام سے اور اس کو ویڈوے دی 20 کی پھر ہم نے دوبارہ x کا ایک اور ویریبل بنایا جس کو ہم نے less than کا سائز لگا کر ایک ویڈوے دی جو کہ میں نے 10 دی دے جس سے اس کو ڈکھانے کا کہ اگر x کی ویڈو 10 سے چھوٹی ہو تو اس میں ایک کی کمی کرو۔ اب یہ لوپ بار بار چکر لگائے گا۔ جس کے بعد ہم یہ رزلٹ ملے گا۔

987654321

**while loop**

ہماری لوپ کی کاس کا دوسرا لوپ ہے۔ یہ بھی فور لوپ کی طرح ہی ہے لیکن اس میں ہم ویریبل کو باہر ہی ڈیفائل کرنے کرتے ہیں۔ اور کنڈیشن کو ہم اسٹیٹمنٹ کے نیچے کی جانب لکھتے ہیں۔ while loop اس طرح لکھا جاتا ہے۔

while () {}

اس کو تفصیل سے سمجھنے کے لئے آپ نیچے درج کوڈ کیجیس پھر میں آپ کو اس کو ڈکھاتا ہوں۔

```
var x=1;
while (x<10) {
document.write(x);
x++;
}
```

ہم نے سب سے پہلے ایک ویریبل بنایا x کے نام سے اور اس کو ویڈوے دی 1 کی پھر ہم نے ایک while loop بنایا اور x کے

ویریبل کو اس میں ڈینائیں کیا جس کو ہم نے greater than کا سائز لگا کر ایک ویلودے دی جیسا کے میں نے یہاں 10 دی ہے جس سے اس کو ڈکھانے کا گر x کی ویلودے 10 سے چھوٹی ہو تو اس میں ایک کا اضافہ کرو۔ اب یہاں بھی ہم وہی کام کریں گے جو ہم نے فور لوپ کے اندر کیا تھا ایک document.write کا استعمال کریں گے جس میں ہم نے x کو لکھ دیا۔ اب ہم DicrementIncrement کرنا ہے جیسا کہ ہم نے فور لوپ کے اندر کیا تھا لیکن یہاں پر یہ اشیاء کے اندر ہو گا جیسا کہ آپ کو ڈی میں دیکھ رہے ہیں۔ میں نے x کے بعد Increment کے سائز document.write کے بعد لکھے ہیں۔ اب یہ لوپ کام شروع کر چکا ہے جو لگاتار چلتے رہیں گے۔ جب تک کہ کنڈیشن غلط نہیں ہو جاتی۔ یعنی کے جیسے ہی ہمارا 1 نمبر 10 کے برابر ہو جائے گا کنڈیشن غلط ہو جائے گی اور زلٹ پنٹ ہونا بند ہو جائے گا اور لوپ نگ بند ہو جائے گی۔ اب جب آپ رزلٹ دیکھیں گے تو آپ کو 1 سے لے کر 9 تک نمبر زمیں گے۔ اس کا رزلٹ بھی اس وقت فور لوپ جیسا ہی ہو گا۔

اس کا رزلٹ یہ ہو گا۔

123456789

### do while loop

ماہری لوپ کی کلاس کا تیسرا لوپ ہے۔ اس لوپ میں بھی ہم ویریبل کو باہری ڈینائیں کرتے ہیں لیکن اس میں ہم do while loop کے بعد دو کری بریکلیس { } لگاتے ہیں اور اس کے اندر ہم کو ڈکھتے ہیں۔ پھر ہم while لکھتے ہیں اور پھر ہم دو راؤنڈ بریکلیس ( ) لگاتے ہیں۔ اور اس کے اندر ہم کنڈیشن لکھتے ہیں۔ اس لوپ میں اور for loop اور while loop میں صرف اتفاق ہے کہ اگر کنڈیشن کی ویلوا اور ویریبل کی ویلوا یک جیسی بھی ہوں تو بھی یہ کم از کم ایک دفعہ رزلٹ ضرور پنٹ کرتا ہے لیکن فور لوپ یا loop کے اندر ایسا نہیں ہوتا اگر ان میں دونوں ویلوا یک ہو جائیں تو رزلٹ نہیں آتا لیکن اس میں آ جاتا ہے۔ اس طرح لکھا جاتا ہے۔

do { } while ()

اس کو تفصیل سے سمجھنے کے لئے آپ نیچے درج کوڈ دیکھیں پھر میں آپ کو اس کو ڈکھاتا ہوں۔

```
var x=1;
```

```
do {
```

```
document.write(x);
```

```
x++;
```

```
} While (x<10)
```

ہم نے سب سے پہلے ایک ویریبل بنایا x کے نام سے اور اس کو ویلودے دی 1 کی پھر ہم نے do لکھنے کے بعد دو کرنی بریکٹس کا استعمال کیا ہے جس کے اندر ہمیں اپنا کوڈ لکھنا ہے فور لوپ اور loop کی طرح ہم یہاں بھی document.write کا ہی استعمال کر رہے ہیں کیونکہ ہمیں یہاں بھی اپنے ویریبل x کو پرنسٹ کروانا ہے۔ پھر ہم نے اپنے ویریبل x کو Increment کا سائز لگا کر تاتا دیا کہ ہمیں اپنے x کے ویریبل میں ایک کا اضافہ چاہیے۔ اب ہم نے while لکھا اور اس کے آگے دو اونڈ بریکٹس لگادے جس میں اب ہم کنڈیشن لکھیں گے اور ہم نے جو کنڈیشن دی ہے اس سے اب یہاں بھی بلکل دوسرے لوپس کی طرح ہی چلے گے۔

اس کا رزولٹ یہ ہو گا۔



```
123456789
```

لیکن جیسا کے میں نے پہلے شروع میں بتایا تھا کہ اگر ویریبل کی ولیو اور کنڈیشن کی ولیو ابرابر ہو جائے تو بھی یہ رزولٹ ضرور پرنسٹ گے گا۔ اب کوڈ دیکھیں۔

```
var x=10;
```

```
do {
```

```
document.write(x);
```

```
x++;
```

```
} While (x<10)
```

یہاں آپ دیکھ سکتے ہیں کہ ہم نے ویریبل کو بھی 10 کی ولیووڈی اور کنڈیشن میں بھی بتا دیا کہ اگر x کی ولیو 10 سے کم ہو تو ہی اس میں 1 کا اضافہ کرو۔ لیکن یہاں پر اس میں کوئی فرق نہیں پڑے گا اور ہمارے پاس کم از کم ایک مرتبہ رزلٹ ضرور آئے گا جو کہ آپ نیچے دیکھ سکتے ہیں۔

اس کا رزلٹ یہ ہو گا۔

10

ایسا اس لئے ہوا ہے کہ do پہلے کوڈ کو یکسی کیوڈ کرتا ہے یعنی پہلے یہ ایک کا اضافہ کر رہا ہے پھر while لوپ کو چلا رہا ہے۔ جس سے ہمیں کم از کم ایک مرتبہ رزلٹ ضرور مل رہا ہے۔ اگر میں ویریبل کی ولیو 10 کی جگہ 9 کرو دتا تو بھی ہمیں صرف ایک ہی رزلٹ ملتا۔

### Switch & Case

ایک قسم کا لوپ ستم ہے۔ جس کے اندر ہم Break کی کمانڈ بھی استعمال کرتے ہیں لوپ کو ختم کرنے کے لئے۔

اسے سمجھنے کے لئے نیچے درج کوڈ دیکھیں۔

```
var name = 'asif';

switch (name) {
    case 'asif' : document.write ('Hello Asif');

    break;
}
```

اس کوڈ کے اندر ہم نے سب سے پہلے ایک name کے نام سے ویریبل بنایا جس کو ولیو دے دی ہم نے asif کی اور پھر ہم نے switch کی کمانڈ استعمال کی جس کے آگے دو راؤنڈ بریکسٹ کے اندر ہم نے اپنے ویریبل کا نام لکھ دیا اور اس کے بعد ہم نے دو کری بریکسٹ لگائے اور اس کے اندر ہم نے case کا استعمال کیا ہے۔ یہاں ہم نے case کوڈ اس ریکٹ asif کا نام دے دیا اور اس کے

بعد ہم نے ایک semi-colon لگایا اور پھر ہم نے document.write کا استعمال کیا یہاں میں بتاتا چلوں کہ آپ یہاں document.write کی جگہ alert کا استعمال بھی کر سکتے ہیں جیسے آپ چاہیں۔ آگے چلتے ہیں ہم نے document.write کے اندر Hello Asif کیونکہ ہم چاہتے ہیں ہے Hello Asif پر نہ ہو۔ پھر ہم نے اس کے آگے break; کی کمائڈ کا استعمال کیا جس سے ہمارا کوڈ یہاں ختم ہو گیا۔ اب یہاں ہو گایہ کہ اگر ہمارے ویریبل کی ولیو asif ہے تو ہم نے case میں جو ڈائریکٹ asif کا تھا وہ یقینی ہو جائے گی اور ہمیں اپنے پیچ پر Hello Asif نظر آئے گا۔ اس کا رезولٹ یہ ہو گا۔

### Hello Asif

**نوت:** ہم نے case کے اندر asif کا چھوٹا لکھا ہے جب کہ document.write کے اندر Asif بڑا ہے کیونکہ وہ ہمیں پر نہ کروانا تھا لیکن Case کے اندر وہ ویریبل کی ولیو ہے اگر آپ نے اس میں کچھ گزبری کی وہ رزلٹ نہیں آئے گا۔

اب آگے بڑھتے ہیں اور اس کو اگلے لیول پر لے جاتے ہیں۔ یہاں ہم نے تین case کیے ہیں اور ہر case کے بعد break کی کمائڈ کو بھی استعمال کیا ہے اور یہ ضروری بھی ہے۔ ورنہ رزلٹ نہیں ملے گا۔

```

name = 'asif';

switch (name) {

    case 'asif' : document.write ('Hello Asif');

        break;

    case 'Danish' : document.write ('Hello Danish');

        break;

    case 'Junaid' : document.write ('Hello junaid');

        break;

}

```

اوپر جو ہم نے تین case لکھے ہیں سب کی ویواہگ ہیں۔ اب جو بھی ویواہارے ویریبل سے میچ ہو گی ہمیں اس case کا پرنٹ ملے گا۔ لیکن اگر کوئی بھی ویواہ نہ ہوئی تو یہ ابھی کوئی رزلٹ نہیں دے گا اور Error کا سبب بننے گا۔ اس کام کے لئے ہم default کی مانڈ کا استعمال کرتے ہیں جیسے نیچے درج مثال میں ہم نے ویریبل کی ویواہ asif سے تبدیل کر کے ahsan کر دی ہے۔ جو کہ ہمارے کسی بھی case سے میچ نہیں ہوتی۔ تو ہم نے default کا استعمال کیا ہے نیچے دیکھیں۔

```

name = 'ahsan';

switch (name) {
    case 'asif' : document.write ('Hello Asif');

        break;

    case 'Danish' : document.write ('Hello Danish');

        break;

    case 'Junaid' : document.write ('Hello junaid');

        break;

    default : document.write ('There is no match');

}

```

اب یہاں ہم نے default کے بعد وہی سب کچھ کیا ہے جو ہم نے case میں کیا تھا اور there document.write کے اندر there defult کو پرنٹ کروادیا جس کے بعد اگر کوئی ویواہ نہ ہو تو رزلٹ یہ ہو گا۔ is no match

There is no match

default کے بعد break کی مانڈ کی کوئی ضرورت نہیں رہتی۔ case اور switch کو مزید سمجھنے کے لئے اگلی مثال کی طرف چلتے ہیں۔ جس سے آپ کو یہ مزید واضح ہو جائے گا۔

```

switch (new Date().getDay()) {

    case 0: document.write("Today is Sunday");
        break;

    case 1: document.write("Today is Monday");
        break;

    case 2: document.write("Today is Tuesday");
        break;

    case 3: document.write("Today is Wednesday");
        break;

    case 4: document.write("Today is Thursday");
        break;

    case 5: document.write("Today is Friday");
        break;

    case 6: document.write("Today is Saturday");
        break;
}

```

اس مثال میں ہم نے date کے فٹشزر کا استعمال کیا ہے۔ new Date() اور getDay() و فٹشزر کا استعمال کیا ہے اور دونوں کو سکنیڈنٹ کرنے کے لئے ایک dot کا استعمال کیا ہے۔ date کے بارے میں ہم آگے مزید تفصیل سے پڑھیں گے۔ اس کے اندر ہم نے جس() کا استعمال کیا ہے اس سے ہمارے پاس اب ہفتے کے ساتوں دن آگئے ہیں۔ جو کنہر زکے حساب سے ہوتے ہیں۔ جو 0 سے شروع ہوتے ہیں اور 6 پر ختم ہوتے ہیں۔ ہم نے ہر case کو ہر دن کا ایک نمبر دے دیا جس کے بعد document.write کا استعمال کرتے ہوئے ہم نے ہر case کے ساتھ ایک پیغام پرنسٹ کروادیا۔ یہاں ہم نے ہفتے کے نام ہی پرنسٹ کروانے ہیں اپ کچھ بھی پرنسٹ کرو سکتے ہیں۔ آج چونکہ ہفتے کا دن ہے اسی لئے آج ہمارے پاس یہ رزلٹ آئے گا۔

**Today is Saturday**

آپ اسے اپنی ویب سائٹ پر لگا سکتے ہیں جس سے جو بھی یوزر آپ کی ویب سائٹ پر آئے گا۔ اسے ہر دن ایک نیا پیغام ملے گا۔ آپ اس میں مہینے اور سال کا بھی کر سکتے ہیں۔

### Return

جیسا کہ اس کے نام سے ہی ظاہر ہوتا ہے کسی چیز کو واپس کرنے کو Return کہتے ہیں۔ میں بھی ہم JavaScript کو کوئی ایک ویڈو دیتے ہیں اور وہ ویڈو وقت ضرورت ہمیں Return کرتا ہے۔ اس کو ہم ویریبل سے کمپیوٹر کر سکتے ہیں۔ اس کو سمجھنے کے لئے ہم ایک فنکشن بناتے ہیں myfunction کے نام سے اور اس میں return کو ایک ویڈو دیں گے۔ کوڈ دیکھیں۔

```
function myfunction () {
    return 'Getting return';
}
```

یہاں ہم نے ایک فنکشن بنایا myfunction کے نام سے اور اس میں return کا استعمال کیا جس کو ہم نے ایک string ویڈو دی getting return کی اب ہونا تو یہ چاہیے کہ جب ہم اس فنکشن کو کال کریں تو مریئن کی ویڈو پرنٹ ہو جائے لیکن ایسا نہیں ہو گا۔ اگر میں صرف myfunction کو کال کروں تو مجھے کوئی رزلٹ نہیں ملے گا۔ مثلاً کوڈ دیکھیں۔

```
function myfunction () {
    return 'Getting return';
}

myfunction();
```

اب اس کوڈ میں ہم نے myfunction کو ڈائرکٹ کال کیا لیکن اس کے باوجود ہمارے پاس کوئی رزلٹ نہیں آئے گا۔ اس کام کے لئے ہمیں اپنے فنکشن کو document.write یا پھر alert کے اندر پرپنٹ کروانا ہو گا۔ ہم یہاں document.write کا

استعمال کریں گے کوڈ دیکھیں۔

```
function myfunction () {
    return 'Getting return';
}

document.write(myfunction());
```

اب ہمیں جو رزلٹ ملے گا وہ یہ ہو گا۔

### Getting return

### Break & Continue

یہ دو کیمیٹریز ہیں Break اور Continue ان دونوں کو سمجھنے کے لئے ہم ایک فور لوپ کا استعمال کریں۔

```
for (x=1; x<10; x++){
    document.write (x+ ' number line <br>');
}
```

یہاں پر ہم نے ایک فور لوپ بنایا جس میں ایک x کے نام سے ویریبل بنایا جو کو ہم نے 1 کی ولیودے دی اور پھر کنڈیشن میں ہم نے بتایا کہ اگر x کی ولیو 10 سے ہم ہوتا تو 1 کا اضافہ کرو جس سے اب جتنی بار لوپ چلے گا اس میں ایک کا اضافہ ہو جائے گا۔ پھر ہم نے اس کو پرنٹ کروانے کے لئے document.write کا استعمال کیا۔ یہاں پر آپ دیکھ سکتے ہیں کہ میں نے x کے بعد ایک + کا سائز لگا کر ایک string ویڈو دی ہے جس کا مقصود آپ کو آسانی سے سمجھانا ہے۔

اس کا رزالٹ یہ ہو گا۔

- 1. number line**
- 2. number line**
- 3. number line**
- 4. number line**
- 5. number line**
- 6. number line**
- 7. number line**
- 8. number line**
- 9. number line**

اب ہم اسی کوڈ کے اندر if کی اسٹیٹمنٹ کا استعمال کریں گے۔ کوڈ دیکھیں۔

```
for (x=1; x<10; x++){
    if (x == 3) {break;}
    document.write (x+ '. number line <br>');
}
```

یہاں پر ہم نے اسی کوڈ کے اندر if کا استعمال کیا ہے جس میں ہم نے بتایا کہ اگر x 3 کے برابر ہو تو برابر ہو تو یہ اسٹیٹمنٹ بریک ہو جائے تو ہمیں جو رزالٹ آئے وہ ملے۔ اب آپ جب کوڈ کو دیکھیں گے تو یہ لوپ 1 اور 2 کے بعد بریک ہو جائے گا اور آگے یہ لوپ چلانا بند ہو جائے گا۔

اس کا رزلٹ یہ ہو گا۔

**1. number line**

**2. number line**

یہ تو کام تھا break کا اور ہم چلتے ہیں continue کی طرف یہاں پہنچی ہم if کا ہی استعمال کریں گے۔ ہم نے اس کوڈ کے اندر بھی بتایا کہ اگر x برابر ہو 3 کے تو یہ اسٹیٹمنٹ continue ہی رہے۔

```
for (x=1; x<10; x++){
    if (x == 3) {continue;}
    document.write (x+ '. number line <br>');
}
```

اس کوڈ سے اب جب ہمارا لوپ 3 تک پہنچے گا تو یہ تین نمبر لائن کو skip کرے گا اور لوپ کو آگے لے کر چل پڑے گا۔

اس کا رزلٹ یہ ہو گا۔

**1. number line**

**2. number line**

**4. number line**

**5. number line**

**6. number line**

**7. number line**

**8. number line**

**9. number line**

## Type Conversion

اس کے اندر ہم JavaScript کے کسی ایک ڈیٹا نیپ کو کسی دوسرے ڈیٹا نیپ میں تبدیل کر سکتے ہیں۔

```
<p id="ctype"></p>
<script>
document.getElementById("ctype").innerHTML =
typeof "Danish";
</script>
```

سب سے پہلے ہم نے ایک p کا نیگ بنایا اور اس کو id="ctype" کی اور اس کی JavaScript کا سارا کام کریں گے۔ اس کو ڈی میں ہم نے سب سے پہلے document.getElementById کا فونکشن استعمال کیا اور اس میں id="ctype" کی آپ کوئی بھی رکھ سکتے ہیں اس کے بعد ہم نے HTML کا استعمال کیا اس سے ہم HTML کے کسی بھی نیگ کے اندر کے مواد کو تبدیل کر سکتے ہیں۔ اس کو ہم نے ویلودی typeof کی یہ JavaScript کی ایک ماندہ ہے جس سے ہم کسی بھی ڈیٹا کی نیپ معلوم کر سکتے ہیں۔ ہم نے اسے ویلودی Danish جو کہ ایک اسٹرنگ ویلو ہے تو اس کا رزلٹ بھی اسٹرنگ آتا چاہیے۔

اس کا رزلٹ یہ ہے۔

string

اس کے اندر بھی ہم نے وہی تمام کام کئے ہیں جس اس مرتبہ ہم نے ڈیٹا نیپ تبدیل کر دیا ہے اس مرتبہ ہم نے ویلوبر میں دی ہے اس طرح اس کا ڈیٹا نیپ بھی تبدیل ہو جائے گا۔

```
<p id="ctype"></p>
<script>
```

```
document.getElementById("ctype").innerHTML =
    typeof 3.14;
</script>
```

اس کا ریٹن یہ ہے۔

number

اس کے اندر بھی وہی سب کچھ ہے اور اس کو ہم نے ویلودی ہے NAN کی جس سے اس کا ذیل اناپ نمبر آئے گا کیونکہ NAN کی ویلودی پر ڈیفائلٹ طور پر ڈیلٹ اناپ نمبر آتا ہے۔

```
<p id="ctype"></p>
<script>
document.getElementById("ctype").innerHTML =
    typeof NaN;
</script>
```

اس کا ریٹن یہ ہے۔

number

اس کے اندر ہم نے ویلودی boolean کی دی ہے جیسا کہ آپ جانتے ہیں کہ boolean کی ویلودی true یا false ہوتی ہے تو ان میں سے کوئی بھی ویلودی نے پر ڈیلٹ اناپ boolean کی آئے گی۔

```
<p id="ctype"></p>
```

```
<script>
document.getElementById("ctype").innerHTML =
typeof false;
</script>
```

اس کا ریٹن یہ ہے۔

### boolean

اس کے اندر ہم نے ایک Array بنایا ہے لیکن چونکہ Array بنیادی طور پر ایک object ہوتا ہے اس لئے اس کا ذیلی نام پر بھی یہی object ہو گا۔

```
<p id="ctype"></p>
<script>
document.getElementById("ctype").innerHTML =
typeof [1,2,3,4];
</script>
```

اس کا ریٹن یہ ہے۔

### object

یہ ایک object ہے اسی لئے اس کا ذیلی نام پر بھی object ہو گا۔

```
<p id="ctype"></p>
```

```
<script>
document.getElementById("ctype").innerHTML =
    typeof {name:'john', age:34};
</script>
```

اس کا ریٹن یہ ہے۔

### object

اس کے اندر ہم new کا نشان دیا ہے جو کہ بنیادی طور پر JavaScript کا ایک object ہے اسی لئے اس کا ذیلا نام پر object ہو گا۔

```
<p id="ctype"></p>
<script>
document.getElementById("ctype").innerHTML =
    typeof new Date();
</script>
```

اس کا ریٹن یہ ہے۔

### object

اس کے اندر ہم نے ایک نشان بنا یا ہے اسی لئے اس کا ذیل نام پر function ہو گا۔

```
<p id="ctype"></p>
```

```
<script>
document.getElementById("ctype").innerHTML =
typeof function () {};
</script>
```

اس کا ریز لٹ یہ ہے۔

### function

اس کے اندر ہم نے کوئی ویلو نہیں دی ایک اسٹرینگ ویلو myCar دی ہے لیکن ڈبل کوٹس کے بغیر جس سے یہ ایک وری ہیل بن جاتا ہے۔  
لیکن چونکہ اس وری ہیل کو کوئی ویلو ابھی تک نہیں ملی ہے اس لئے اس کا ڈیٹا ناپ undefined ہو گا۔

```
<p id="ctype"></p>
<script>
document.getElementById("ctype").innerHTML =
typeof myCar;
</script>
```

اس کا ریز لٹ یہ ہے۔

### undefined

اس کے اندر ہم نے myCar کی طرح صرف اسٹرینگ ویلو دی ہے null کی لیکن JavaScript کے اندر ڈیفارٹ طور پر null ایک کی ورڈ ہے اس لئے اس کا ڈیٹا ناپ object ہو گا۔

```
<p id="ctype"></p>
<script>
document.getElementById("ctype").innerHTML =
typeof null;
</script>
```

اس کا ریز لٹ یہ ہے۔

### object

#### constructor Property

ہم اس پروپرٹی کا استعمال کسی بھی ڈیٹا تپ کا تعمیری نگاشن معلوم کرنے کے لئے کرتے ہیں یعنی کسی نگاشن کو بنانے کے لئے کس طرح کے نگاشن کی مدد لی جاسکتی ہے۔ کوڈ دیکھیں۔

```
<p id="constr"></p>
<script>
document.getElementById("constr").innerHTML =
"john".constructor
</script>
```

اس کوڈ میں آپ دیکھ سکتے ہیں کہ ہم نے پہلے ہی کی طرح ایک p کا نیگ بنایا اور اس کو id constr کی اس کے بعد script کے اندر JavaScript کا کوڈ لکھا اور اس کوڈ میں ہم نے document.getElementById کے اندر id لکھ دی جو ہم نے پیر گراف کے نیگ کے اندر لکھی تھی۔ اس کے بعد ہم نے .innerHTML کا استعمال کر کے ایک اسٹرینگ ویلوب concadinate constructor کی پروپرٹی کے ساتھ کر دیا۔

اس کا رزلٹ یہ ہے۔

### **function String() { [native code] }**

یعنی string کا فنکشن ہو گا اور native code سے مراد یہاں کوڈ ہو گا۔ جو کچھ اس طرح لکھا جائے گا۔

### **String() {}**

یہ نام فنکشن کی طرح ہی لگ رہا ہے۔ اور کام بھی تقریباً ایسا ہی کرے گا۔ اس کے بارے میں مزید تفصیل سے آگے پڑھیں گے۔ اگر کوڈ میں ہم نے وہی کام دوبارہ کیا اور اس بارہم نے جو ویڈوی ہے وہ ہے نمبر جس سے اب ہمارے پاس فنکشن نمبر کا آئے گا۔

```
<p id="constr"></p>
<script>
document.getElementById("constr").innerHTML =
(3.14).constructor
</script>
```

اس کا رزلٹ یہ ہے۔

### **function Number() { [native code] }**

اس بارہم نے ایک boolean ویڈوی ہے جو کہ false یا true کا مطلب یہ ہے تو boolean کا اس کا مطلب یہ ہوتی ہے۔ ہم نے false کا

```
<p id="constr"></p>
```

```
<script>

document.getElementById("constr").innerHTML =
false.constructor

</script>
```

اس کا ریٹن یہ ہے۔

**function Boolean() { [native code] }**

یہاں پر تم نے ایک Array بنایا ہے اور اس کا جواب ایک Array کا نشان ہو گا۔

```
<p id="constr"></p>

<script>

document.getElementById("constr").innerHTML =
[1,2,3,4].constructor

</script>
```

اس کا ریٹن یہ ہے۔

**function Array() { [native code] }**

یہاں ایک object ہے تو جواب میں ہمیں ایک object کا نشان لے گا۔

```
<p id="constr"></p>

<script>
```

```
document.getElementById("constr").innerHTML =
    {name:'john', age:34}.constructor
</script>
```

اس کا ریٹن یہ ہے۔

### **function Object() { [native code] }**

یہاں پر ہم نے new Date() کا استعمال کیا ہے۔ یہ ہے تو بنیادی طور پر ایک object لیکن یہ ہمیں Date کا نتائج دے گا کیونکہ Date کا عالمی نتائج دے نتائج ہوتا ہے۔

```
<p id="constr"></p>
<script>
document.getElementById("constr").innerHTML =
    new Date().constructor
</script>
```

اس کا ریٹن یہ ہے۔

### **function Date() { [native code] }**

اس بارہم نے چونکہ Date() کا ہی استعمال کیا ہے اس لئے جواب میں بھی ہمارے پاس function ہی آئے گا۔

```
<p id="constr"></p>
<script>
```

```
document.getElementById("constr").innerHTML =
function () {}.constructor
</script>
```

اس کا رزلٹ یہ ہے۔

**function Function() { [native code] }**

### Converting Numbers to Strings

یہاں پر ہم نمبرز کو اسٹرینگ ویلو کے اندر تبدیل کریں گے۔ اس کام کے لئے ہم نے ایک p کا نیگ کے اندر numstr کی id بنائی۔

```
<p id="numstr"></p>
```

اس کے بعد ہم نے script کے نیگ کے اندر ایک ویریبل بنایا x کے نام سے اور اس کو ایک نمبر ویلو دے دی 123 کی اس کے بعد ہم document.getElementById کا استعمال کرتے ہوئے پیر آگراف کے نیگ کی id کی numstr نے جو ہم نے لکھی تھی دے دی اس کے بعد ہم نے String کے اندر HTML کے innerHTML کا استعمال کرتے ہوئے ویریبل x کو اس کے اندر لکھ دیا۔ اس سے اب x کی ویلو جو کہ نمبر ہے string کے ساتھ اب string ویلو بن گئی ہے۔

```
<script>
var x = 123;

document.getElementById("numstr").innerHTML = String(x);

</script>
```

اس کا رزلٹ یہ ہے۔

123

اس مثال میں ہم نے وہی کام کیا یعنی اس مرتبہ ویریبل بنانے کے بعد نمبرو بیلو کے ذائز کرت String کے فنکشن کے اندر لکھ دیا۔

```
<p id="numstr"></p>
<script>
document.getElementById("numstr").innerHTML = String(123);
</script>
```

اس کا رезولٹ یہ ہے۔

123

اس مثال میں ہم نے پچھلی مرتبہ کی طرح ایک نمبرو بیلو دی اور اس مرتبہ ہم نے اس String کے فنکشن کے اندر جمع بھی کر دیا۔

```
<p id="numstr"></p>
<script>
document.getElementById("numstr").innerHTML = String(100 + 23);
</script>
```

اس کا رезولٹ یہ ہے۔

123

## Date and Time

JavaScript کے اندر ہم تاریخ اور وقت بھی اپنے چیز کے اندر show کر سکتے ہیں۔ اور اس کا استعمال کر کے ان سے مختلف کام بھی لئے جاسکتے ہیں۔ JavaScript کے اندر کوئی بھی تاریخ یا وقت شروع کرنے سے پہلا ہمیں ایک بنیادی فنگشن کا استعمال کرتا پڑتا ہے۔ جس کے لئے ایک فنگشن new Date(); کا استعمال ہوتا ہے۔ اس فنگشن سے ہی باقی کے تمام فنگشن ڈیٹا اٹھاتے ہیں اور ہمیں مطلوبہ معلومات دے دیتے ہیں۔ یہی درج اس کا Syntax ہے۔

**new Date();**

اس کو دیکھیں کہ یہ کس طرح کا output دیتا ہے۔ یاد رکھیں آپ جیسا دیکھ رہے ہیں آپ کو ویسا ہی لکھنا ہے۔ اگر الفاظ چھوٹا ہیں تو چھوٹا لکھیں اور اگر بڑا ہیں تو بڑا لکھیں۔ فنگشن میں ذرا بھی فرق آپ کے لئے مشکل پیدا کر سکتا ہے۔ اور Error کا سبب بن سکتا ہے۔ اس کو بلکل اسی طرح لکھنا ہے اگر ذرا بھی فرق ہو گا تو یہ کام نہیں کرے گا اور Error کا سبب بنے گا۔

```
var date = new Date();  
document.write (date);
```

ہم نے پہلے ایک ویریبل بنایا date کے نام سے پھر اس میں ہم نے new Date(); کا فنگشن استعمال کیا اور پھر ہم نے date کے ویریبل کو اپنے چیز میں پرنٹ کر دیا۔

اس کا رезولٹ یہ ہے۔

**Sun Sep 20 2015 22:28:21 GMT+0500 (Pakistan Standard Time)**

اس کو ڈسے آپ کو ساری چیزیں ایک ہی جگہ دیکھ جاتی ہیں۔ اگر ان میں سے کوئی ایک چیز مثلاً صرف وقت چاہیے تو آپ مزید فنگشن کا استعمال کرتے ہیں۔ جن کے بارے میں ہم آگے تفصیل سے پڑھیں گے۔ اس کو ڈکھانے کے چار طریقے ہیں۔ پہلا آپ دیکھ چکے

بیں جس میں صرف `newDate()` کے نتائج کا استعمال ہوا ہے۔ بقایا کی طرف بھی ایک نظر ڈالتے ہیں پھر آگے بڑھتے ہیں۔  
نیچے درج اس کا Syntax ہے۔

**`new Date (year, month, day, hours, minutes, seconds, milliseconds);`**

اس نتائج میں 7 پیرامیٹر (سال، مہینہ، دن، گھنٹے، منٹس، سیکنڈز، ملی سیکنڈز) ہوتے ہیں۔ یہ امر کمی فارمیٹ ہے اور ان کو اسی ترتیب سے لکھا جاتا ہے۔ JavaScript کے اندر سال 1 جنوری 1970 سے شروع ہوتا ہے۔ لیکن ہم ان پیرامیٹر کا استعمال کر کے کوئی بھی تاریخ اور وقت حاصل کر سکتے ہیں۔ نیچے کوڈ دیکھیں۔ لیکن چونکہ JavaScript کے اندر مہینہ، دن، گھنٹے، منٹ، سیکنڈز اور ملی سیکنڈز سب 0 سے شروع ہوتے ہیں۔ اسی لئے اس کے بعد دن پھر گھنٹے،

```
var date = new Date (1991,2,06,01,30,30,0);
document.write (date);
```

اس کوڈ میں ہم نے `new Date()` کے اندر سال جو کہ ہم یہاں 1991 دے رہے ہیں۔ پھر اس کے بعد مہینہ جو کہ مارچ کا ہے اور اس کے لئے ہم نے 2 دیا ہے کیونکہ صفر سے شروع ہونے پر 2 نمبر مارچ کا ہوگا۔ اس کے بعد ہم نے تاریخ رکھی ہے جو کہ چھتاریخ ہے۔ اس کے بعد ہم نے گھنٹے میں 1 اور منٹ میں 30 منٹ کا وقت رکھا ہے۔ سیکنڈز میں 30 سیکنڈز اور ملی سیکنڈز کو صفر رکھا ہے۔ ویسے سیکنڈز اور ملی سیکنڈز کو نہ بھی دیں تو کوئی فرق نہیں پڑے گا۔ لیکن آپ لوگوں کو سمجھنا نے کے لئے لکھ دیا ہے۔ اب ہو گایہ کہ ہمارے `newDate()` کے نتائج میں ہم نے جو پیرامیٹر دیے ہیں یہ اس دن کا بلکل صحیح حساب لگا کر اس دن کی صحیح تاریخ ہمیں دے دے گا۔ مثلاً نیچے رزلٹ میں آپ دیکھ سکتے ہیں کہ دن بدھ آرہا ہے لیکن ہم نے نہیں بتایا کہ دن کون سا ہے اور فارمیٹ بھی وہ نہیں جو ہم نے دیا تھا۔ ہونا تو یہ چاہئے تھا کہ پہلے سال آتا پھر مہینہ پھر دن اور پھر گھنٹے، منٹ اور سیکنڈز وغیرہ۔ لیکن اس کے بر عکس یہاں ہمارے پاس پہلے ہفتہ کا دن آرہا ہے اس کے بعد مہینہ پھر تاریخ پھر سال اور پھر وقت دے رہا ہے اور ساتھ ہی ساتھ یہ ہمیں ہمارے کمپیوٹر کے وقت سے ہمیں یہ بتا رہا ہے کہ یہ پاکستان کا استینڈرڈ نام ہے۔ اس کا رزلٹ یہ ہو گا۔

**Wed Mar 06 1991 01:30:30 GMT+0500 (Pakistan Standard Time)**

اس اگلی مثال سے آپ کو مزید واضح ہو جائے گا۔ یہاں ہم صرف تاریخ کے متعلق کام کر رہے ہیں اور وقت کا اس سے کوئی لیندا و یانہیں نہ ہے۔ یہاں بھی ہم نے وہی پیرامیٹر زاستعمال کئے ہیں۔ یعنی پہلے سال پھر مہینہ اور آخر میں تاریخ دی ہے۔ یہچکے کوڈ دیکھیں۔

```
var date = new Date (1991,2,06);  
document.write (date);
```

اس کا آٹھ پٹ یہ ہو گا۔ اس میں کچھ خاص فرق نہیں پڑا لیکن صرف وقت رات کے 12:00 بجاءے ہیں۔ اور پوچیں گھنٹے کے فارمیٹ کی وجہ سے یہ 00:00:00 نظر آ رہا ہے۔

**Wed Mar 06 1991 00:00:00 GMT+0500 (Pakistan Standard Time)**

یہ ( ) newDate کو لکھنے کا ایک اور طریقہ کار ہے۔ اس میں ہم میں سینڈز کا استعمال کرتے ہیں۔ یہ کافی مشکل ہے لیکن کبھی کبھی کام آتا ہے۔ ایک دن میں 86,400,00 میں سینڈز ہوتے ہیں۔

یہچکے درج اس کا Syntax ہے۔

### **new Date (milliseconds)**

چونکہ JavaScript کا سال 1 جنوری 1970 سے شروع ہوتا ہے اسی لئے اگر آپ صرف 86,400,00 میں سینڈز لکھتے ہیں تو آپ کو وہی تاریخ ملے گی جس کے کمپیوٹر میں وقت UTC UTC اسٹینڈرڈ نام کے مطابق اس وقت کیم جنوری 1970 ہو گا اور دن جمعرات کا ہو گا اور وقت دونج کر چوپیں منٹ ہو رہے ہونگے۔ لیکن اگر آپ کے کمپیوٹر میں اپنے علاقہ کا نام سیو ہوا جیسے میرے کمپیوٹر میں پاکستان کا وقت سیو ہے تو اس صورت میں میری طرح آپ کے پاس بھی یہی رزلٹ آئے گا جو میرے پاس آ رہا ہے۔

```
var date = new Date (8640000);
```

`document.write (date);`

اس کا ریز لٹ پاکستان اسٹینڈرڈ نام کے مطابق یہ ہوگا۔

**Fri Jan 02 1970 05:00:00 GMT+0500 (Pakistan Standard Time)**

اور اس کا ریز لٹ UTC اسٹینڈرڈ نام کے مطابق یہ ہوگا۔

**Thu Jan 01 1970 02:24:00 GMT+0000 (Coordinated Universal Time)**

اب آجاتے ہیں ڈیٹ اسٹرینگ کی طرف اس کا استعمال گھری یا کیانڈر بنانے میں ہوتا ہے۔ ڈیٹ اسٹرینگ دراصل ایک فنکشن ہے جو `newDate ()` کے ساتھ کام کرتا ہے۔ یچے درج اس کا Syntax ہے۔

**`new Date (dateString)`**

یچے درج کوڈ کے اندر ہم نے `d` کے نام سے ایک ویریبل بنایا جس کو ہم نے `newDate()` کی مانند دی اس کے بعد ہم نے ایک اور `d` کا ویریبل بنایا اسی نام سے کنفیوزن ہوں JavaScript کے اندر ہم ایک نام کے مزید ویریبلوں بھی بناسکتے ہیں یہ عام بات ہے۔ اب ہم نے اپنے اگے ویریبل کو پہلے والے `d` کے ویریبل کے ساتھ `toString()` کے ساتھ کنکیڈ نہ کر دیا جس کے بعد ہمارے پہلے والے `d` کے ویریبل کی ساری چیزیں اب اگے `d` کے ویریبل کے پاس آگئی۔ اب ہو گا یہ کہ پہلے `d` کا ویریبل `newDate()` کے ساتھ منگش کے ساری تاریخ اٹھائے گا اور پھر اسے اگے `d` کے ویریبل کے اندر `toString()` کے ساتھ متحمل کرائے جو میڈر کار ہے دے گا اور بعد میں `document.write()` ہماری مطلوبہ `d` کی دیلوں جو `d` کا ویریبل ہمیں دے رہا ہے پہنچ کر دے گا۔ اس کو مزید سمجھنے کے لئے یہ کوڈ دیکھیں۔

`d = new Date();`

`d = d.toString();`

```
document.write( d );
```

اس کا ریز لٹ ایسا ہو گا۔

**Mon Sep 21 2015 06:54:45 GMT+0500 (Pakistan Standard Time)**

اس اگے کوڈ میں بھی ہم وہی کام کر رہے ہیں۔ سارا سب کچھ وہی ہے جس میں `toString()` کے لئے `Date` کے `toString()` کا استعمال کر رہے ہیں۔ جس سے اب ہمارے پاس صرف مہینہ، دن، تاریخ اور سال آئے گا۔ اس کو مزید سمجھنے کے لئے نیچے کوڈ دیکھیں۔

```
d = new Date();
d = d.toDateString();
document.write( d );
```

اس کا ریز لٹ ایسا ہو گا۔

**Mon Sep 21 2015**

یہ بھی ویسا ہی کوڈ ہے اس سے ہمیں UTC نام ملتا ہے۔ اس کو مزید سمجھنے کے لئے نیچے کوڈ دیکھیں۔

```
d = new Date();
d = d.toUTCString();
document.write( d );
```

اس کا ریز لٹ ایسا ہو گا۔

Mon, 21 Sep 2015 01:52:37 GMT

اس کوڈ سے ہمیں ہمارے کمپیوٹر کا صرف وقت ملتا ہے اس کا زیادہ استعمال گھری بنانے میں ہوتا ہے اس کام کے لئے ہمیں صرف `toLocaleTimeString()` کے فنگشن کا استعمال کرنا ہوتا ہے۔ اس کو زید سمجھنے کے لئے نیچے کوڈ دیکھیں۔

```
d = new Date();
d = d.toLocaleTimeString();
document.write(d);
```

اس کا رезالٹ ایسا ہو گا۔

6:56:55 AM

## Date

اب یہاں تک توبات ہو گئی؛ `newDate()` کے فنگشن کی بنیاد کی لیکن جیسا کہ آپ دیکھ سکتے ہیں اس میں ہمیں اب بھی تاریخ، دن، سال، مہینہ، منٹ، سینڈ زاویلی سینڈ زسب ساتھ ہی مل رہے ہیں۔ لیکن اگر ہمیں سب کو الگ الگ کرنا ہو تو ہم کیا کریں گے یا یوں کہیں کے ہمیں اس میں سے صرف ایک چیز درکار ہے جیسے اگلی مثال میں ہمیں وقت درکار ہے تو ہم کیا کریں اب اس کام کے لئے ہم آچکے ہیں `get` اور `set` کی طرف سب سے پہلے `get` کی طرف چلتے ہیں۔

سب سے پہلے ہمیں تاریخ چاہئے اس کام کے لئے ہم `( )` `getDate()` کا ایک فنگشن استعمال کریں گے۔ یہ دو الفاظ `get` اور `Date` کا مجموعہ ہے۔ جن کے ملنے سے یہ فنگشن بناتے۔ یہ JavaScript کے اندر `newDate()` سے صرف مہینہ کے دن اختاتا ہے اور ہم اس کا استعمال صرف اپنے پیچ میں مہینہ کے دن شو کرنے کے لئے یا مہینہ کے دنوں سے متعلق کسی پروگرامنگ کے لئے کرتے ہیں۔ اس کے دو پیر امیز ہوتے ہیں جو کہ 1 سے لے کر 31 تک ہوتے ہیں جن سے ہم اپنی مرضی کی تاریخ بھی تیار کر سکتے ہیں۔

```
var date = new Date ();
var Date = date.getDate ();
document.write(Date);
```

یہاں پر آپ دیکھ سکتے ہیں کہ میں نے ایک ویریبل بنایا date کے نام سے پھر اس کے اندر میں نے new Date{} کا فنگشن استعمال کیا۔ اس سے اب یہ ہمیں پوری تاریخ ایک ساتھ مل گئی۔ پھر اس کے بعد ہم نے ایک اور ویریبل بنایا Date کے نام سے اور اس کو ہم نے date کے ویریبل کی دلیل دے دی۔ یعنی ہم نے پہلے والے ویریبل کو دوبارہ دوسرے والے ویریبل کے اندر دوبارہ استعمال کیا۔ جس سے اب ہم اس کے اندر سے کچھ بھی نکال سکتے ہیں۔ جیسا کہ آپ جانتے ہیں کہ جاوائسکریپٹ ایک Case Sensitive زبان ہے اور اس میں ایک حرف کا فرق بھی اہمیت رکھتا ہے۔ اسی لئے یہ ایک مثال بھی ہے اور آپ دیکھ سکتے ہیں کہ ہم نے ایک ہی نام کے دو ویریبل بنائے صرف پہلے کا d چھوٹا تھا اور دوسرے کا D بڑا اور اس سے بھی یہ دو الگ الگ ویریبل بن گئے۔ اور اس کا م کے لئے ہم get کا فنگشن استعمال کریں گے۔ لیکن جیسا کہ میں پہلے بھی بتاچکا ہوں کہ ہم ایک سے زیادہ فنگشنز کو یڈ کرنے کے لئے dot(.) کا استعمال کرتے ہیں۔ اسی لئے ہم یہاں بھی ایسا ہی کرنے والے ہیں۔ اس کا M کے لئے ہم نے date کے بعد ایک (.dot) لگایا اور پھر (); getDate کا فنگشن استعمال کیا۔ یہاں ہم دونوں کے بارے میں معلوم کرنا چاہتے ہیں۔ اب اس کوڈ سے ہمارے پاس صرف اس دن کی تاریخ آئے گی جو ہمیں چاہیے۔

اس کا رезولٹ یہ ہو گا۔

اس مثال میں ہم سال کے مہینوں کے بارے میں معلوم کریں گے۔ اس کا M کے لئے بھی ہم new Date () کے فنگشن کا استعمال کریں گے کیونکہ یہ بنیاد ہے اسی سے ہمیں ساری تاریخ ملتی ہے۔ اب مہینے کے بارے میں معلوم کرنے کے لئے ہم پھر اس طریقے کا استعمال کریں گے جس کا استعمال ہم نے مہینے کے دن معلوم کرنے کے لئے کیا تھا۔ لیکن یہاں ہم مہینے کے لئے

`getMonth()` کا استعمال کریں گے۔ اس کے بھی دو پیرامیٹر ہوتے ہیں جو کہ 0 سے لے کر 11 تک ہوتے ہیں جن سے ہم اپنی مرضی کی تاریخ بھی تیار کر سکتے ہیں۔ نیچے درج کوڈ دیکھیں پھر تفصیل میں آتے ہیں۔

```
var date = new Date () ;  
  
var month = date.getMonth ()+1 ;  
  
document.write(month) ;
```

یہاں بھی آپ دیکھ سکتے ہیں کہ ہم نے `date` کے ویریبل کے اندر `(newDate())` کا فنکشن استعمال کیا پھر ایک اور ویریبل بنایا `month` کے نام سے اور اس میں ہم نے `date` کے ویریبل کے ساتھ `(getMonth())` کا فنکشن کو کوئی بینٹ کر دیا۔ ابھی تو یہ صرف مہینہ کا نمبر ہی پرنٹ کرے گا۔ لیکن آپ `switch Case if else` کا استعمال کر کے انہیں `String` میلوں میں بھی پرنٹ کرو سکتے ہیں۔ اس کوڑ کے استعمال سے آپ مہینہ معلوم کر سکتے ہیں۔ لیکن اس میں ایک مشکل یہ ہے کہ JavaScript کے اندر مہینے، اور وقت کے گھنٹے، منٹ، سینٹز وغیرہ سب کے سب صفر 0 سے شروع ہوتے ہیں۔ جس کا مطلب ہے کہ اگر مہینہ ساتواں ہے تو آپ کا جواب 6 آئے گا۔ مثلاً دسمبر کا مہینہ بارہواں ہوتا ہے لیکن JavaScript کے اندر یہ گیارہوں نمبر پر آئے گا اور جنوری پہلا ہوتا ہے لیکن یہ صفر نمبر پر آئے گا۔ اس کا حل یہ ہے کہ آپ اس کے اندر ایک عدد جمع کر دیں۔ لیکن یاد رکھیں کہ جیسا لکھا ہے آپ کو بلکل ویسا یہ لکھنا کوڑ نہ کوڑ غلط ہو گا۔

اس کا راست یہ ہو گا۔

سال کے لئے بھی ہم اسی طریقہ کار کا استعمال کریں گے جس کا استعمال ہم نے دن اور مہینے کے لئے کیا تھا۔ اس کے چار پیرامیٹر ہوتے ہیں جو کہ چار ہندسوں پر مشتمل ہوتے ہیں۔ جن سے ہم اپنی مرضی کی تاریخ بھی تیار کر سکتے ہیں۔ یہاں ایک بات اور ذہین رکھ لیں کہ JavaScript کے اندر سال 1970 سے ہوتا ہے۔ اس سے پہلے کا سال آپ کو خود بنانا پڑے گا۔

```
var date = new Date();
var year = date.getFullYear();
document.write (year);
```

یہاں بھی آپ دیکھ سکتے ہیں کہ ہم نے date کے ویریبل کے اندر; (newDate) کا فنگشن استعمال کیا پھر ایک اور ویریبل بنایا year کے نام سے اور اس میں ہم نے date کے ویریبل کے ساتھ; (getFullYear) کا فنگشن کو کونکیڈ یعنی کوڈ لکھا ہے۔ یہ بھی دوسرے فنگشن کی طرح ہی کام کرے گا۔ اس کوڑ کے استعمال سے ہمیں سال مل جائے گا۔ لیکن یاد رہے کہ جیسا ہم نے کوڈ لکھا ہے ویسا ہی لکھنا ہے ورنہ غلط ہو گا۔ سال کے اس کوڑ کا استعمال آپ اپنی ویب سائٹ میں بھی کر سکتے ہیں جس سے آپ کی ویب سائٹ میں سال خود بخوبی دیکھ جو جائے گا۔

اس کا رزلٹ یہ ہو گا۔

2015

Day

اسی طرح ہفتے کے دنوں کے لئے بھی ہم ایک فنگشن; (getDay) کا استعمال کریں گے۔ اس کے سات پیروی امیز ہوتے ہیں جو کہ 0 سے لے کر 6 تک ہوتے ہیں جس سے ہم اپنی مرضی کے دن کی معلومات حاصل کر سکتے ہیں اور اس کا استعمال بھی کر سکتے ہیں۔  
نیچے درج کوڑ دیکھیں پھر تفصیل میں آتے ہیں۔

```
var date = new Date();
var day = date.getDay();
document.write (day);
```

ہر بار کی طرح یہاں بھی آپ دیکھ سکتے ہیں کہ ہم نے `newDate()` کے ویریبل کے اندر `date` کا فنکشن استعمال کیا پھر ایک اور `getDay()` کے نام سے اور اس میں ہم نے `date` کے ویریبل کے ساتھ `getDay()` کے فنکشن کو کوئیڈینٹ کر دیا۔ اس سے اب ہمیں ہفتے کا جو دن ہو گا وہ مل جائے گا۔ اس میں بھی ہفتے کا دن 0 سے شروع ہوتا ہے۔ جن کے بارے میں آپ `switch Case` میں پہلے ہی دیکھ چکے ہیں۔ یعنی اگر نمبر 6 ہو گا تو دن ہفتے کا ہو گا۔ اس کا رزلٹ یہ ہو گا۔

6

Hour

اب آتے ہیں گھنٹے کی طرف ایک دن میں چوبیس گھنٹے ہوتے ہیں۔ اور گھنٹوں کو ویب پیچ پر شو کرنے کے لئے بھی ہم ایک فنکشن `getHours()` کا استعمال کرتے ہیں۔ اس کے بھی دو پیرامیٹر ہوتے ہیں جو کہ 0 سے لے کر 23 تک ہوتے ہیں۔ یہ چوبیس گھنٹے کے فارمیٹ پر ہوتا ہے۔ لیکن ہم `if` اور `else` کا استعمال کر کے اسے AM اور PM میں بھی کنورٹ کر سکتے ہیں۔ نیچے درج کوڈ دیکھیں۔

```
var time = new Date();
var hour = time.getHours();
document.write(hour);
```

اس کوڈ میں ہم نے سب سے پہلے ایک `time` کے نام سے ویریبل بنایا اور پھر اسے `newDate()` کی مانگ دی اس کے بعد ایک اور `getHours()` کے نام سے اور اس کو `time` کے ویریبل کے ساتھ کوئیڈینٹ کر کے `()` کے فنکشن کا استعمال کیا۔ اور `document.write()` کے ذریعے اسے پرنسٹ کر دیا۔

اس کا رزلٹ دیکھیں۔

7

میرے کمپیوٹر میں اس وقت صبح کے ساتھ نج رہے ہیں اسی لئے نمبر سات دے رہا ہے لیکن اگر میرے کمپیوٹر میں دوپہر کے ایک بجے ہوتے تو اس وقت یہ 13 دن تا کیونکہ جیسا میں پہلے ہی بتاچکا ہوں کہ یہ چوبیس گھنٹے کے فارمیٹ پر ہوتا ہے۔ اب اسے 12 گھنٹوں کے فارمیٹ پر لانے کے لئے ہم if اور else کا استعمال کریں گے۔ لیکن پہلے ہم PM اور AM کیسیں گے ورنہ بعد میں ہمارا کوڈ غلط ہو جائے گا۔

```
var time = new Date ( );
var hour = time.getHours();
document.write( hour + " " +sort );
if(hour < 12){
    sort = "AM";
}
else (sort="PM");
```

یہاں ہم نے ایک if کا فنکشن استعمال کی اور اس میں کنڈیشن دی کہ اگر hour کا ویریبل کم ہو 12 سے تو پھر ہم نے ایک sort کے نام سے ایک ویریبل بنایا اور اس میں ایک اسٹرینگ ہیلودے دی AM کی جس سے اب جیسے ہی ہمارا وقت 12 گھنٹوں سے کم ہو گا وہ AM ہو جائے گا۔ اور پھر ہم نے else کا استعمال کی اور اس میں sort کی ویلو PM کرو دی۔ جس سے اب جیسے ہی وقت 12 گھنٹوں سے اوپر جائے گا۔ اس کی ویلو PM ہو جائے گی۔ اور اس کو میں نے document.write کے اندر hour کے ویریبل کے ساتھ پرنٹ کر دیا۔ آپ ان دونوں کے درمیان میں ڈبل کوئی شن اور ان کے درمیان ایک اپسیس دیکھ رہے ہوں گے اس سے ان دونوں کے درمیان تھوڑا فاصلہ جائے گا۔

اس کا ذریث سچھا سطح ہو گا۔

13 PM

یہ غلط ہے 13 بجے PM نہیں آتا۔ اب اس کو 12 گھنٹوں کے فارمیٹ میں تبدیل کرنے کے لئے ہم if کا استعمال کریں گے۔

```
var time = new Date( );
var hour = time.getHours();
document.write( hour + " " + sort );
if(hour < 12){
    sort = "AM";
}
else (sort="PM");
if(hour > 12){
    hour=hour-12;
}
```

ہم نے یہاں ایک if کا فنگشن بنایا اور اس میں کندیشن لکھی کہ اگر hour کا ویریبل 12 سے زیادہ ہو تو hour کے اندر کے ویریبل کوہی اسائیں کرو اور اس میں سے 12 کو ماہنس کرو اس طرح سے اب اور پرواںے ویریبل کا تعلق ختم اب hour کے ویریبل کے پاس جو ویو جائے گی وہ ہمارے if کے فنگشن کے hour کے ویریبل کی ویو جائے گی۔ اب ہمارے پاس جو ویو آئے گی وہ کچھ اس طرح ہوگی۔

1 PM

Minutes

اسی طرح ایک گھنٹے میں 60 منٹ ہوتے ہیں جن کو شو کرنے کے لئے بھی ہم getMinutes() کے فنگشن کا استعمال کریں گے۔ اس کے بھی دو پیرامیٹر ہوتے ہیں جو 0 سے لے کر 59 تک ہوتے ہیں۔

```

var time = new Date ();
var minute = time.getMinutes();
document.write (minute);

```

یہاں آپ دیکھ سکتے ہیں کہ ہم نے پہلے `time` کے نام سے ایک ویریبل بنایا اور اس کو یادوی(); `newDate()` کی پھر ایک اور ویریبل بنایا۔ `minute` کے نام سے اور اس کو `time` کے ویریبل کے ساتھ(); `getMinutes()` کے فنگشن کے ساتھ کلکہ بینڈ کر دیا پھر اس کے بعد ہم نے `document.write` کے اندر `minute` کے ویریبل کو پرنٹ کر دیا۔

اس کا راز لٹ یہ ہو گا۔

5

یہاں آپ دیکھ سکتے ہیں کہ 5 منٹ ہونے ہیں اور صرف 5 نظر آ رہا ہے جو کہ گھری ہیں بلکل بھی اچھا نہیں لگے گا۔ ہمیں 5 کے آگے ایک 0 کی ضروری ہے۔ اس کام کے لئے ہم ایک if کے فنگشن کا استعمال کریں گے۔

```

if (minute < 10){
    minute="0"+minute;
}

```

آپ دیکھ سکتے ہیں کہ ہم نے پہلے if کا ایک فنگشن بنایا جس کو کلکہ بینڈ دی کہ اگر `minute` کا ویریبل کم ہو 10 سے تو اس `minute` کے ویریبل کو 0 کر دیں اور آپ دیکھ سکتے ہیں کہ 0 ایک اسٹرینگ ویلو بنایا ہوا ہے کیوں کہ ہمیں اسے ایسا ہی پرنٹ کروانا ہے۔ اسے اسٹرینگ ویلو بنانے کے لئے ہم نے 0 کو ڈبل کوٹیش کے اندر لکھا ہے۔ پھر ہم نے منٹ کے ویریبل جس کو ہم نے 0 کیا تھا اس کے اندر پھر سے ہم نے `minute` کے ویریبل کا اضافہ کر دیا۔ اب ہو گا یہ کہ جب ہمارا `minute` کا ویریبل پرنٹ ہو گا تو پہلے جسے ہم نے 0 کیا تھا وہ منٹ کو 0 کر دے گا پھر جو ہم نے + کے سامنے کے ساتھ دوبارہ `minute` کے ویریبل کا اضافہ کیا تھا اس سے ہمارے پرنٹ یا گھری میں منٹ کو 0 کر دے گا۔ لیکن اس بارہ شو ہو جائیں گے۔ اس بارہ شو کے آگے ایک 0 بھی ہو گا۔ ہم اس کام کے لئے 0 کو ڈال رکھ بھی لکھو

سکتے تھے لیکن پھر اس صورت میں جب منٹس 10 یا اس سے زیادہ ہوتے اور بھی 0 ان کے آگے ہی شو ہوتا جو بلکل اچھا نہیں لگتا اسی لئے ہمیں یہ کوڈ لکھنا پڑتا۔  
اس کا راز لٹ یہ ہو گا۔

06

Seconds

منٹس کی طرح ایک سینڈ میں بھی 60 ملی سینڈ زہوتے ہیں اور ان کے لئے ہم ;() کے فنکشن کا استعمال کرتے ہیں۔ اس کے بھی دو ہی پیر ایمیٹر زہوتے ہیں جو کہ 0 سے لے کر 59 تک ہوتے ہیں۔

```
var time = new Date ();
var second = time.getSeconds();
document.write (second);
```

یہاں بھی آپ دیکھ سکتے ہیں پہلے والا ہی کوڈ ہے جس minute کے ویریبل کی جگہ second کا ویریبل ہے اور getSeconds() کی جگہ getMinutes(); ہے۔

اس کا راز لٹ یہ ہو گا۔

1

اور یہاں بھی وہی صورت حال ہے سینڈ بھی اکیا ہی آئے گا۔ اس کے آگے ہی 0 کے لئے وہی if کا فنکشن استعمال کرنا ہو گا، اسی باز minute کی جگہ second کا استعمال کرنا ہو گا۔

```

if(second < 10){

second="0"+second;

}

```

اس کا ریز لٹ یہ ہو گا۔

09

### Milli Seconds

ملی سینڈز کے پیرو امیٹر زو تین ہوتے ہیں لیکن یہ ہوتا چار ہندسوں والا ہے۔ ایک سینڈ میں ایک ہزار میٹر ہوتے ہیں۔ اس کے لئے ہم `(getMillisecond() کے) نکشن کا استعمال کریں گے۔ اس کے پیرو امیٹر زو سے لے کر 999 تک ہوتے ہیں۔`

```

var date = new Date ();

var milisecond = date.getMilliseconds();

document.write (milisecond);

```

اس کا ریز لٹ ایسا ہو گا۔ کیونکہ اب جتنی دیر میں ویب سائٹ کو ریفریش کریں گے اتنی دیر میں دوسو یا ڈھانی سو ملی سینڈز گزر چکے ہوں گے۔

563

آپ کی آسانی کے لئے کچھ بنیادی فنکشنز کا ایک چارٹ موجود ہے جس سے آپ کو پریشانی نہیں ہوگی۔

فونکشن	تفصیل
newDate()	مہینے کے دن جو کہ 1 سے لے کر 31 تک ہونگے۔ (1-31)
getDay()	ہفت کے دن جو کہ 0 سے شروع ہو کر 6 تک ہونگے۔ (0-6)
getMonth()	مہینے جو کہ 0 سے 11 تک ہونے۔ (0-11)
getFullYear()	سال جو کہ چار ہندسوں میں ہوگا۔ (1970) (yyyy)
getHours()	گھنٹے جو کہ 0 سے 23 تک ہونگے۔ (0-23)
getSeconds()	سیکنڈ جو کہ 0 سے 59 تک ہونگے۔ (0-59)
getMinutes()	منٹ جو کہ 0 سے لے کر 59 تک ہونگے۔ (0-59)
getMilliseconds()	میلی سیکنڈ جو کہ 0 سے لے کر 999 تک ہونگے۔ (0-999)
getTime()	وقت جو کہ JavaScript کے شروع وقت یعنی 1 جنوری 1970 سے ہوگا۔ جس میں امریکی اسینڈرڈ کے مطابق مہینہ پہلا پھر دن اور آخر میں سال آئے گا۔ (January 1, 1970)

## Cookie

یہاں ہم Cookie کے بارے میں پڑھیں گے۔ یہ کہانے والی Cookie نہیں بلکہ اس کا استعمال ہم یادداشت کے طور پر کرتے ہیں۔ یہ بھی دراصل BOM کا حصہ ہے۔ cookie کے دو بنیادی کام ہوتے ہیں۔ پہلا یوزر کی انفارمیشن حاصل کرنا اور دوسرا جب یوزر دوسری بار پیچ و زٹ کرتے تو اسے یاد رکھنا۔ مثال کے طور پر جب آپ فیس بک یا اپنی کسی ای میل پر لوگ ان کرتے ہیں تو یوزر دوسری بار پیچ و زٹ کرتے ہے اگر آپ اسے سائیکل کر لیتے ہیں تو وہ آپ کے کمپیوٹر میں ایک فائل بنادیتی ہے اور اگر آپ لوگ آٹھ نکریں تو جب آپ دوبارہ وہ فائل و زٹ کرتے ہیں تو آپ کو دوسری بار لوگ ان کرنے کی ضرورت نہیں رہتی وہ دوبارہ ایسے ہی اپن ہو جاتی ہے۔ cookie ایک مخصوص وقت کے لئے ہوتی ہے اور پھر اس کے بعد خود بے خود ذیلت ہو جاتی ہے۔

cookie کو بنانے کے لئے ہم document.cookie کا استعمال کرتے ہیں۔

مثال:

```
document.cookie="username=Danish Irshad; expires=Thu, 18 Dec 2015
12:00:00 UTC";
```

cookie کو پڑھنے کے لئے بھی ہم document.cookie کا استعمال کرتے ہیں۔

مثال:

```
var x = document.cookie;
document.write( d );
```

cookie کو تبدیل کرنے کے لئے بھی ہم document.cookie کا استعمال کرتے ہیں۔ اس سے تبدیل شدہ cookie پر انیسونیتی کو حفظ کرنے کے لئے بھی ہم document.cookie کا استعمال کرتے ہیں۔ اس سے تبدیل شدہ cookie پر جاتی ہے۔

مثال:

```
document.cookie="username=document.cookie=username=Muhammad
Danish Irshad; expires=Thu, 18 Dec 2015 12:00:00 UTC";
```

cookie کو ختم کرنے کے لئے بھی ہمیں صرف اس کی expire date کو تبدیل کرنا ہے۔

مثال:

```
document.cookie = "username=; expires=Thu, 01 Jan 1970
00:00:00 UTC";
```

## DOM

DOM یا Document Object Model دراصل ویب میں موجود اکیومنٹس کے لئے ہوتے ہیں۔ ویب پیچ جیسے ہی لوڈ ہوتا ہے یہ اپنا کام شروع کر دیتا ہے۔ عام طور پر اس کو کسی لفظ کو تبدیل کرنے یا پرنٹ کروانے کے لئے استعمال کیا جاتا ہے۔ ہم اس کو دو حصوں میں تقسیم کر سکتے ہیں۔ مثال کے لئے آپ نیچے والا کوڈ دیکھیں۔

```
<p id="demo">Hello</p>
<script>
document.getElementById("demo").innerHTML = "Hello World";
</script>
```

اس کوڈ کے اندر ہم نے پہلے ایک پیر اگراف کا نیگ استعمال کیا جس کو ہم نے id "demo" کے نام سے پھر ہم نے id "demo" کے اندر اپنا کوڈ لکھا۔ پہلے ہم نے document.getElementById() کا کوڈ استعمال کیا جس سے ہمارے براوزر کو پتا چلتا ہے کہ اب ہمیں ایک پتا کرنا ہے کہ کس نیگ کی id "demo" کا نام کو id "demo" کا کوڈ استعمال کیا جس کو ہم نے ویلودے دی "Hello World" کی اسے ایک ویریبل کی طرح ہی ویلودی جاتی ہے۔ کوڈ دیکھیں۔

"Hello World";

اور جیسے کہ پہلے بھی پڑھ چکے ہیں ان دونوں کوڈز کو ملانے کے لئے ہمیں dot(.) کا استعمال کرنا پڑتا ہے تو ہم نے ان دونوں کوڈز کے درمیان ایک dot(.) لاگا دیا۔

اب پیر اگراف کے اندر Hello World کو یہ کوڈ دیکھیں۔

اب جو زیادہ تر لوگوں کے ذہن میں سوال آتا ہے کہ یہ کام تو ہم ڈائریکٹ بھی کر سکتے تھے تو اتنی محنت کیوں؟

تو جناب جواب یہ ہے کہ اس کا استعمال ہم مختلف پوزیشن پر کرتے ہیں۔

DOM کے مزید Elements یہ ہیں ان سے آپ کو اس کو سمجھنے میں آسانی ہو گی۔  
یہ کہاتے ہیں۔ Finding Elements

### **document.getElementById()**

اس کوڈ سے ہم HTML ٹیگ کی id کو معلوم کر کے اس پر کام کرتے ہیں۔ اس کے بارے میں ہم پہلے ہی پڑھ چکے ہیں۔

### **document.getElementsByTagName()**

اس کوڈ سے ہم HTML ٹیگ کا نام لکھتے ہیں اور پھر اس پر کام کرتے ہیں۔ لیکن اگر آپ کے پیچ میں ایک ہی نام کے بہت سارے ٹیگ ہیں تو یہ سب پر کام کرے گا اسی لئے اس کو بہت کم استعمال کیا جاتا ہے۔

### **document.getElementsByClassName()**

اس کوڈ سے ہم HTML ٹیگ کی Class کو معلوم کر کے اس پر کام کرتے ہیں۔ یہ id کی ہی طرح کام کرتا ہے۔

یہ کہاتے ہیں۔ Changing Elements

### **.innerHTML=**

اس کا استعمال بھی ہم پہلے ہی کر چکے ہیں۔ یہ HTML کے اندر کسی بھی ٹیگ کے مداخل کو تبدیل کرنے کے لئے استعمال ہوتا ہے۔

### **.attribute=**

اس کا استعمال بھی ہم پہلے ہی کرچکے ہیں۔ یہ HTML کے اندر کسی بھی نیگ کے مواد کو تبدیل کرنے کے لئے استعمال ہوتا ہے۔

### **.setAttribute(attribute,value)**

اس کا استعمال بھی ہم پہلے ہی کرچکے ہیں۔ یہ HTML کے اندر کسی بھی نیگ کے مواد کو تبدیل کرنے کے لئے استعمال ہوتا ہے۔

### **.style.property=**

اس کا استعمال ہم JavaScript کے اندر CSS کی پروپرٹی کی استعمال کے لئے کرتے ہیں۔ اس کی پروپرٹی مختلف ہو سکتی ہیں جن کو **=** کے سامنے کے بعد لکھتے ہیں۔

مثال کے طور پر **.style.color=**, **.style.font=** وغیرہ کہلاتے ہیں۔ Adding and Deleting Elements

### **document.createElement()**

اس کا استعمال بھی ہم پہلے ہی کرچکے ہیں۔ یہ HTML کے اندر کسی بھی نیگ کے مواد کو تبدیل کرنے کے لئے استعمال ہوتا ہے۔

### **document.removeChild()**

اس کا استعمال بھی ہم پہلے ہی کرچکے ہیں۔ یہ HTML کے اندر کسی بھی نیگ کے مواد کو تبدیل کرنے کے لئے استعمال ہوتا ہے۔

### **document.appendChild()**

اس کا استعمال بھی ہم پہلے ہی کر چکے ہیں۔ یہ HTML کے اندر کسی بھی ٹیگ کے مواکو تبدیل کرنے کے لئے استعمال ہوتا ہے۔

### **document.replaceChild()**

اس کا استعمال بھی ہم پہلے ہی کر چکے ہیں۔ یہ HTML کے اندر کسی بھی ٹیگ کے مواکو تبدیل کرنے کے لئے استعمال ہوتا ہے۔

### **document.write()**

اس کا استعمال بھی ہم پہلے ہی کر چکے ہیں۔ یہ JavaScript کے اندر کسی بھی مowaکو یہ چیز پر پرنٹ کرنے کے لئے استعمال ہوتا ہے۔

## **BOM**

درالی براوزر یا BOM (Browser Object Model) کو سات حصوں میں تقسیم کر سکتے ہیں۔ جن میں سے دو حصوں Popup Boxes اور Cookies کے بارے میں ہم پہلے ہی پڑھ چکے ہیں اب مزید پانچ کے بارے میں پڑھیں گے۔

## **Window**

وند و اسکرین کا استعمال ہم JavaScript کے اندر وند کے لئے کرتے ہیں اس کے چند کوڈ زدرجہ ذیل ہیں۔ ویسے تو popup وند وہ window prefix کے بغیر کہی جاتی ہیں اسی لئے انہیں الگ کیفگری میں رکھا جاتا ہے۔

### **window.open()**

اس سے ہم نئی وند و اوپن کرنے کے کام آتا ہے۔

**window.close()**

اس سے ہم اپن وندو کو بند کرنے کے کام آتا ہے۔

**window.moveTo()**

اس سے ہم اپن وندو کو موو (move) کرنے کے کام آتا ہے۔

**window.resizeTo()**

اس سے ہم اپن وندو ری سائز یعنی چھوٹا بڑا کرنے کے کام آتا ہے۔

**window.innerHeight**

اس سے ہم نئی اپن وندو کی اوپرائی معلوم کر سکتے ہیں۔

**window.innerWidth**

اس سے ہم نئی اپن وندو لمبائی معلوم کر سکتے ہیں۔

**document.body.clientHeight**

اس سے ہم ویب پیچ کی اوپرائی معلوم کر سکتے ہیں۔

**document.body.clientWidth**

اس سے ہم ویب سائٹ کی لمبائی معلوم کر سکتے ہیں۔

**Window Screen**

اس کا استعمال ہم اسکرین کی معلومات اور سائز گر کے لئے کرتے ہیں۔

**screen.width**

اس سے ہم اسکرین کی لمبائی معلوم کر سکتے ہیں۔

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML = "Screen Width: " +
screen.width;
</script>
```

رزائلٹ

**Screen Width: 1280**

**screen.height**

اس سے ہم اسکرین کی اوپرائی معلوم کر سکتے ہیں۔

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML = "Screen Height: " +
screen.height;
</script>
```

رزٹ

**Screen Height: 1024**

**screen.availWidth**

اس سے ہم یا اسکرین کی موجودہ لمبائی معلوم کر سکتے ہیں۔

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML = "Available Screen Width: " +
screen.availWidth;
</script>
```

رزٹ

**Available Screen Width: 1280**

**screen.availHeight**

اس سے ہم اسکرین کی موجودہ اونچائی معلوم کر سکتے ہیں۔

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML = "Available Screen Height: "
+ screen.availHeight;
</script>
```

رزولٹ

Available Screen Height: 995

**screen.colorDepth**

اس سے ہمیں پتا لگتا ہے کہ ہماری اسکرین پر کتنے کلر گتے ہیں ایک کلر کوش کرنے کے لئے مثلاً 24 بٹ میں 16,777,216 مختلف کلر ہوتے ہیں۔ جبکہ 32 بٹ میں 4,294,967,296 مختلف Deep کلر ہوتے ہیں۔ True

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML = "Screen Color Depth: " +
screen.colorDepth;
```

&lt;/script&gt;

رزملٹ

**Screen Color Depth: 24****screen.pixelDepth**

اس پروپرٹی کا استعمال ہم اسکرین کے پکسل ڈپٹھ کے معلوم کرنے کے لئے کرتے ہیں۔

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML = "Screen Pixel Depth: " +
screen.pixelDepth;
</script>
```

رزملٹ

**Screen Pixel Depth: 24****Window Location**

وندلوکیشن کا استعمال ہم موجودہ چیز کا URL یا اس چیز کو دوسرا جگہ ری ڈائیریکٹ کرنے کے لئے کرتے ہیں۔

**window.location.href**

اس کا استعمال ہم چیز کا URL معلوم کرنے کے لئے کرتے ہیں۔

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML =
"Page location is " + window.location.href;
</script>
```

رزک

**Page location is**

**/F:/Danish%20Files/19-August-2015js/js\_window\_location.htm**

**window.location.hostname**

اس کا استعمال ہم چیز کا Hostname معلوم کرنے کے لئے کرتے ہیں۔

مثال:

```
<p id="demo"><p>
<script>
```

```
document.getElementById("demo").innerHTML =
"Page hostname is " + window.location.hostname;
</script>
```

رزٹ

**Page hostname is**

**window.location.pathname**

اس کا استعمال ہم یقیناً کا پاتھ معلوم کرنے کے لئے کرتے ہیں۔

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML =
"Page path is " + window.location.pathname;
</script>
```

رزٹ

**/19-August-2015/js\_window\_location.htm**

**window.location.protocol**

اس کا استعمال ہم یقین کا پروگرام معلوم کرنے کے لئے کرتے ہیں۔

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML =
"Page protocol is " + window.location.protocol;
</script>
```

رزملٹ

**Page protocol is file:ftp://**

### Window History

وندو ہسٹری کا استعمال ہم وندو کی ہسٹری کے لئے کرتے ہیں۔

**history.back()**

اس کا استعمال ہم وندو کی ہسٹری سے اپنے یقین پر پیچھے جانے کے لئے کرتے ہیں۔

مثال:

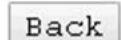
```
<script>

function goBack() {
    window.history.back()
}

</script>

<input type="button" value="Back" onclick="goBack()">
```

رزٹ



### **history.forward( )**

اس کا استعمال ہم وندو کی ہٹری سے اپنے یقین پر آگے جانے کے لئے کرتے ہیں جس چیز کو پہلے ہی وزٹ کر چکے ہوں اور پھر پیچھے جا چکے ہوں۔

مثال:

```
<script>

function goForward() {
    window.history.forward()
}

</script>

<input type="button" value="Forward" onclick="goForward()">
```

رزٹ

[Forward](#)

## Window Navigator

وندو Navigator کے استعمال سے ہم برااؤز کی انفارمیشن حاصل کرتے ہیں۔

### **Navigator Cookie Enabled**

اس کوڈ کے استعمال سے ہمیں پتا چلتا ہے کہ یوزر کے برااؤز میں cookie enable ہے یا نہیں۔

بہت ساری ویب سائٹس کوکی کا استعمال کرتی ہیں۔ اس کوڈ کے استعمال سے اگر برااؤز میں cookie enable نہ ہو تو یہ کوڈ بتا دیتا ہے اور اس صورت میں یوزر کو آپ اپنی ویب سائٹ پر آنے سے روک سکتے ہیں یا اس کے لئے مخصوص چیزیں رکھ سکتے ہیں۔

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML =
"Cookies Enabled is " + navigator.cookieEnabled;
</script>
```

رزک

**Cookies enabled is true**

**Browser Names**

اس کا استعمال کر کے ہم براوزر کا نام معلوم کر سکتے ہیں کہ یوزر کون سا برائزر استعمال کر رہا ہے۔

**نوت:** appName Netscape کا Safari اور Internet Explorer 11, Chrome, Firefox, ہو گا۔

appCodeName Mozilla اور Opera اور Chrome, Firefox, Internet Explorer, Safari, ہو گا۔

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML = "Name is " +
navigator.appName + ". Code name is " + navigator.appCodeName;
</script>
```

رزٹ

**Name is Netscape**  
**Code name is Mozilla**

### Browser Engine

اس سے ہم براوزر کے انجن کے بارے میں معلومات حاصل کرتے ہیں۔

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML = navigator.product;
</script>
```

رزک

### Browser engine is Gecko

### Browser appVersion

اس سے ہم براوزر کے appVersion کے بارے میں معلومات حاصل کرتے ہیں۔

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML = navigator.appVersion;
</script>
```

رزک

5.0 (Windows NT 5.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/46.0.2490.71 Safari/537.36

## Browser userAgent

اس سے ہم براوزر کے userAgent کے بارے میں معلومات حاصل کرتے ہیں۔

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML = navigator.userAgent;
</script>
```

رزک

**Mozilla/5.0 (Windows NT 5.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/46.0.2490.71 Safari/537.36**

## Browser Platform

اس سے ہمیں پتا چلتا ہے کہ براوزر کس اور پینگ سسٹم پر چل رہا ہے۔

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML = navigator.platform;
```

&lt;/script&gt;

رزٹ

**Win32****Browser Language**

اس سے ہمیں پتا چلتا ہے کہ ہمارے براوزر کی زبان اس وقت کون سی استعمال ہو رہی ہے۔

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML = navigator.language;
</script>
```

رزٹ

**Browser language is en-US****Java Enabled**

اس سے ہمیں پتا چلتا ہے کہ ہمارے براوزر میں Java Enable ہے یا نہیں۔

مثال:

```
<p id="demo"><p>
<script>
document.getElementById("demo").innerHTML = navigator.javaEnabled();
</script>
```

رزولٹ

Java enabled is true

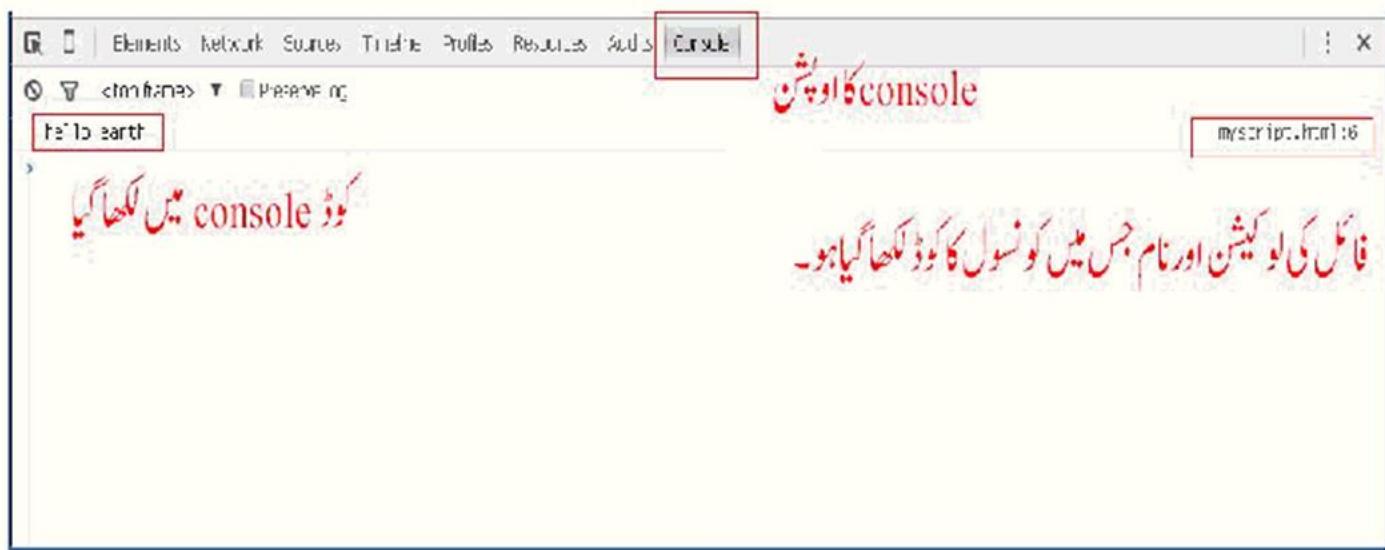
### Console.log();

اس کے استعمال سے ہم براوزر کے console میں کچھ بھی لکھ سکتے ہیں۔

اسے دیکھنے کے لئے آپ internet explorer یا firefox، chrome کا بُٹن دبائیں اور پھر F12 کا بُٹن دبائیں اور پھر اپشن سائیکل کریں آپ کو لکھا دکھانے گا۔

```
console.log("hello earth");
```

اس کا رزلٹ یہ ہو گا۔



فائل کی لوکیشن اور نام جس میں کو نسل کا کوڈ لکھا گیا ہو۔

## Timing Events

نائمنگ انوٹس کا استعمال ہم مختلف کاموں میں کرتے ہیں۔ یہاں ہم setTimeout() اور clearTimeout() کے بارے میں پڑھیں گے۔

اس کے استعمال سے ہم کسی بھی فٹشن کو ایک مخصوص وقت پر چالاسکتے ہیں اس کا وقت ہمیں ملی سینڈز میں دینا ہوتا ہے۔ جیسے 1000 میں ایک سینڈ ☆  
یہ سیکھنے کی طرح ہے لیکن یہ فٹشن ہو لگاتا رہتا ہے۔ setInterval()

setInterval() کے دو پیر امیٹر ہوتے ہیں ایک میں ہم فٹشن کا نام لکھتے ہیں اور دوسرا میں ملی سینڈز لکھتے ہیں۔ اس کو مزید سیکھنے کے لئے کوڈ دیکھیں۔

```
<button onclick="setTimeout(myFunction, 3000);>
click and wait 3 Seconds</button>
```

یہاں ہم نے ایک بٹن بنایا اور اس کے اندر Attribute کا استعمال کیا اور اس کو ہم نے setTimeout() کی ویلو دی اور ہم نے Function کا نام جو کہ ہمارے پاس myFunction کے نام سے ہے لکھا اور دوسرا میں 3000 ملی سینڈز لکھ دیا۔ اور detail میں click and wait 3 Seconds لکھ دیا۔

```
<script>
function myFunction() {
document.write('Hello World');
}
</script>
```

اس لائن میں ہم نے Script کے اندر ایک فنکشن بنایا myFunction کے نام سے اور اس کے اندر ایک document.write کا فنکشن استعمال کیا اور اس کے اندر ہم نے ایک اسٹرینگHello World پرینٹ کروایا۔ اب اس سے ہو گا یہ کہ جب کوئی یوزر بٹن پر کلک کرے گا اور 3 سینڈز را تظاہ کرے گا تو اسے 3 سینڈز کے بعد پر Hello World کا پیغام ملے گا۔

اس کا راز لمحہ یہ ہو گا۔

کلک کے بعد	کلک سے پہلا
Hello World	click and wait 3 Seconds

اس مثال میں ہم نے دو بٹن بنائیں ہیں ایک سے ہمارا فنکشن چلے گا اور دوسرا سے ہمارا فنکشن بند ہو جائے گا۔ فنکشن بند کرنے کے لئے ہم clearTimeout کا استعمال کریں گے۔

```
<button onclick="myVar = setTimeout(myFunction, 3000)">Start</button>
<button onclick="clearTimeout(myVar)">Stop</button>
```

یہاں ہم نے دو بٹن بنائیں ہیں پہلا والے بٹن میں onclick کے event کے اندر ایک ویریبل بنایا myVar کے نام سے اور اس کو 3000 میلی سینڈز دے دیا اور settimeout کی اور اس کے اندر فنکشن کا نام myFunction اور start کا details attribute کے اندر start کھڑا جس سے ہمیں پتا چلے گا کہ اس سے ہمارا فنکشن start ہو گا۔ دوسرا والے بٹن میں ہم نے clearTimeout کا استعمال کیا ہے اور اس کے اندر ہم نے جو ویریبل پہلا والے بٹن میں myVar کے نام سے اسی ویریبل کا نام دے دیا اور details کے اندر Stop دے دیا جس سے ہمیں پتا چلے گا کہ اس سے فنکشن بند ہو گا۔

```
<script>
function myFunction() {
```

```
alert("Hello");
```

```
}
```

اس کے بعد ہم نے ایک فنکشن بنایا myFunction کے نام سے اور اس میں ایک alert کا استعمال کیا ہے۔ اس سے اب یہ ہو گا کہ جب کوئی یوزر ہمارے Start کے بٹن پر کلک کرے گا تو اسے ہر 3 سینڈز کے بعد ایک الٹ ملے گا جب تک کوہ Stop پر کلک نہ کرو۔۔۔

اس کا رزالٹ یہ ہو گا۔



کوہم clock کے باب میں پڑھیں گے۔

## Clock

ہم یہاں آپ کو دو حصوں میں گھری بنا سکھائیں گے۔ پہلے میں ہم بنیادی چیزوں کے بارے میں پڑھیں گیا اور سیکھیں گے۔ اور دوسرے میں ہم اس گھری کو چلا سکھیں گے۔ جس کے لئے ہم HTML اور CSS کا بھی استعمال کریں گے۔

گھری نمبر 1

```
var time = new Date();
```

```

var hour = time.getHours();
var min = time.getMinutes();
var sec = time.getSeconds();

document.write ("Current time is: "+<b>"+hour+":"+min+":"+sec+"</b>");

</script>

```

اس گھنٹی میں ہم نے سب سے پہلے ایک ویریبل بنایا time کے نام سے اور اس کو ڈیلوڈے دی; (new Date() کی یہ ہماری گھنٹی کو وہ تمام چیزیں دے گا جن کی ہمیں ضرورت ہے۔ اس کے بعد ہم نے ایک اور ویریبل بنایا hour کے نام سے اور اس کو time کے ویریبل کے ساتھ concatenate کر کے getHours() کا فنگشن دے دیا۔ یہ ہماری گھنٹی کو گھنٹے دے گا۔ اس کے بعد ہم نے ایک اور ویریبل بنایا min کے نام سے اور اس کو بھی time کو ڈیلوڈے دی concatenate کر دیا۔ اس کے بعد ہم نے ایک اور ویریبل بنایا sec کے نام سے اور اس کو بھی time کے getMinutes() کا فنگشن کی جس سے یہ ہمیں اپنی گھنٹی میں منٹس دے گا۔ اس کے بعد ہم نے ایک اور ویریبل بنایا sec کے نام سے اور اس میں بھی وہی طریقہ کا استعمال کر کے getSeconds() کا فنگشن کو بطور ڈیلوڈے دیا جس سے اب ہمیں ہماری گھنٹی میں سینڈز بھی مل گئے۔

آخر میں ہم نے ان سب کو document.write() کے اندر پرنسٹ کر دیا۔ یہاں اب آپ دیکھ سکتے ہیں کہ ہم نے <br> کا لیگ بھی استعمال کیا ہے؛ document.write() کے اندر اور ایک اسٹرینگ و ڈیلوڈی دی ہے "Current time is: " کی اور ان سب کو جوڑنے کے لئے ہم نے + کے ساتھ کا استعمال کیا ہے یہ ہم نے اس لئے کی ہے کہ ہماری گھنٹی کا وقت bold نظر آئے۔

اس کا رезالٹ یہ ہے۔

Current time is: 5:44:1

لیکن اس میں آپ دیکھ سکتے ہیں کہ سینڈز 1 سے شروع ہو رہے ہیں اور اس کے آگے PM یا AM بھی نظر نہیں آ رہا کیونکہ یہ اس وقت 12 گھنٹے کی گھنٹی ہے اور ڈینالٹ طور پر JavaScript کے اندر 0 نہیں آتا اس کام کے لئے گھنٹی نمبر 2 دیکھیں۔ اور بھی یہ چلے گی بھی نہیں ہمیں یہی کو بار بار refresh کرنا پڑے گا۔

```

var time = new Date ();

var hour = time.getHours();

var min = time.getMinutes();

var sec = time.getSeconds();

```

یہاں آپ دیکھ سکتے ہیں کہ یہ کوڈ اور پروگرام ہی ہے۔ ہمیں بلکل ایسا ہی لکھنا ہے۔

```

document.write ("Current time is: "+"<b>" +hour+ ":" +min+ ":" + " sec "+"  

"+sort+"</b>");
```

لیکن یہاں ہم نے کوڈ میں sort کے نام سے ایک ویریبل کا نام لکھا ہے یہ AM اور PM کے لئے ہے۔ نیچے کوڈ دیکھیں۔

```

if (hour<12){  

    sort="AM";  

} else{sort="PM";}

```

یہاں ہم نے سب سے پہلے ایک if کے نشان کے اندر کا اندھارہ دی کہ اگر hour کا ویریبل less than 12 سے تو sort کے ویریبل کے اندر اس کی ویلو کر دو AM اور else کے اندر sort کی ویلو PM کرو۔ اب اس سے ہو گا یہ کہ جب ہمارے گھنے 12 سے ہم ہونگے تو یہ AM دے گا اور جب وہ 12 گھنٹوں سے اور پر جائیں گے یعنی جب 13 بجے گیں تو PM ہو گا۔

```

if(hour>12){  

    hour=hour-12;  

}

```

اب یہاں آپ دیکھ سکتے ہیں کہ ہم نے if کے نکشیں کو دو بارہ استعمال کرتے ہوئے کامنڈ دی کے کہ hour کا ویریبل زیادہ ہو 12 سے تو hour کے اندر دی hour کے ویریبل میں 12 کو نہیں کر دو۔

نوٹ: اس کوڈ میں پہلے والا hour کا ویریبل اس کی ولیوں والے hour سے مختلف ہے اور پہلے والا ہی گنا جائے گا۔  
اب اس کوڈ سے ہو گایہ کہ جیسے ہی ہماری گھری میں 13 بجے کا وقت ہو گا وہ پھر سے 1 سے شروع ہو جائے گا۔

```
if (min<10){  
    min="0"+min;  
}
```

اب اگلے کوڈ کی طرف آتے ہیں اس کوڈ میں ہم نے if کے ویریبل کے اندر کامنڈ دی کہ اگر min کا ویریبل کم ہو 10 سے تو min کے ویریبل کی ولیوں 0 کر دو اور پھر min کا ویریبل ہی اس میں دوبارہ جمع کر دو۔ اس سے ہو گایہ کہ جیسے ہی منٹس 1 سے شروع ہونگے تو 1 کے آگے 0 لگ جائے گا جو 9 تک چلے گا اور وہ پھر خود بے خود ختم ہو جائے گا۔

```
if (sec<10){  
    sec="0"+sec;  
}
```

سینکنڈز کے لئے بھی ہم نے min کا ہی طریقہ استعمال کے کیا ہے۔ اس سے بھی جب سینکنڈز 1 سے شروع ہونگے تو 1 کے آگے 0 آجائے گا اور 9 کے بعد خود بے خود ختم ہو جائے گا۔

اس گھری کارزنٹ یہ ہو گا۔

6:02:02 AM

لیکن ابھی بھی اس کو refresh کرنا پڑے گا چنانے کے لئے اب اس کو سمجھنے کے لئے ہمیں ایڈ و انس گھٹری میں جانا ہو گا۔

### ایڈ و انس گھٹری نمبر 1

یہاں سب سے پہلے ایک فونکشن بنایا (myTime) کے نام سے اور اس میں document.write() کا استعمال کیا اور اپنی تیار کی ہوئی گھٹری کی تمام چیزیں اس میں لکھ دی اور وہ یہ بدل کر باہر لکھ دیا۔ اور if else کے لئے تمام فونکشن کے اندر ہی لکھ دیے۔

```

var time = new Date ();

var hour = time.getHours();

var min = time.getMinutes();

var sec = time.getSeconds();

function myTime() {

document.write ("Current time is: "+"<b>" +hour+ ":" +min+ ":" + sec +""

"+sort+"</b>");

if (hour<12){

    sort="AM";

} else{sort="PM";}

if(hour>12){

hour=hour-12;

}

if (min<10){

```

```

min="0"+min;
}

if (sec<10){

sec="0"+sec;

}

```

اس کے بعد ہم نے ایک ویریبل بنایا myVar کے نام سے اور اس میں setInterval() کے فناش کا استعمال کریں گے۔ اس میں ہم نے فناش لکھ کر اس کا نام دے دیا() اور myTime() اور 1000 میلی سینڈز دے دیا جس سے اب یہ ہر 1 سینڈ میں چلے گا۔

```
var myVar = setInterval(function(){myTime()},1000);
```

اس کا رезولٹ یہ ہو گا۔

6:45:33 AM

ایڈوانس گھری نمبر 2

```
<div id="clock"></div>
```

اس گھری میں ہم نے سب سے پہلے ایک div کا نیگ بنایا اور اس کو id دے دی clock کی اور اب اسی id پر ہم اپنا CSS اور JavaScript کا کام کریں گے۔ JavaScript کا کوڈ دیکھیں۔

```
<script>
```

```
var myVar = setInterval(function(){myTime()},1000);
```

```

function myTime() {
    var d = new Date();
    document.getElementById("clock").innerHTML =
        d.toLocaleTimeString();
}

</script>

```

سب سے پہلے ہم نے setInterval() کے تکشیں کا استعمال کیا۔ اس میں ہم نے تکشیں لکھ کر اس کا نام دے دیا () اور 1000 میلی سینکنڈز دے دیا جس سے اب یہ ہر 1 سینکنڈ میں چلے گا۔ اس کے بعد ہم نے ایک تکشیں بنایا () کے نام سے اس کے اندر ہم نے ایک ویریبل `new Date()` کے نام سے اور اس کو دیا یہ دی جس سے اب یہ ویریبل ساری انفارمیشن جو ہمیں چاہیے رکھ لے گا۔ پھر اس کے بعد ہم نے document.getElementById() کا استعمال کیا اور جو id ہم نے دی تھی `clock` کے نام سے وہ اس کے اندر لکھ دی۔ اس کے بعد ہم نے `.innerHTML` کی کمائڈ کا استعمال کیا اور اس کے اندر d کے ویریبل کو connect() کر کے concatenate() کے ساتھ `toLocaleTimeString()` کو دیا جس سے اب یہ مقامی وقت حساب سے چلے گا۔

اس کا رезولٹ یہ ہو گا۔

# 6:45:33 AM

اس کے بعد ہم یہاں CSS کا استعمال کریں گے۔ جس سے ہماری گھڑی مزید خوبصورت اور لوگوں کی بن جائے گی۔

CSS کا کوڈ ہم style کے میگ کے اندر لکھتے ہیں۔

```

#clock { background:red; color:yellow; width:82px; font-family:arial;
border:2px dashed green; padding:1px; }

```

ہم نے اس کو ڈی میں background کا رنگ لال رکھا ہے اس کے فونٹ کا فلکر پیلا، اس کی لمبائی 82px، فونٹ کا اسٹائل arial رکھا ہے اس کا border ڈیش اسٹائل میں رکھا ہے وہ بھی سبز رنگ کا اور آخر میں padding 1px رکھی ہے۔

اس سے اب ہماری گھری پچھا اس طرح کی نظر آئے گی۔



اسی طرح کے مزید کوڈ استعمال کر کے آپ مزید گھریاں بھی بناسکتے ہیں۔

6:27:01 AM

6:47:50 AM

### Calculator

ہم یہاں آپ کو ایک بنیادی Calculator بنا سکھائیں گے۔ جس میں آپ جمع (+)، کھٹا (-)، تفریق (÷) جس کے سامنے کے لئے ہم سدیش (/) کا استعمال کرتے ہیں۔ اور ضرب (x) کر سکیں گے۔

```
<FORM NAME="Calc"> <TABLE BORDER=4> <TR> <TD>
```

```
    <INPUT TYPE="text" NAME="Input" Size="16">
```

سب سے پہلے ہم نے ایک form کا نیگ لگایا جس کا نام ہم نے Calc رکھ دیا۔ اس کے بعد ہم نے ایک table بنایا جس کا border ہم نے 4 رکھ دیا اس کے بعد ہم نے ایک tr بنایا اور اس tr کے اندر ایک td بنایا اس td کے اندر ہم نے ایک input کا استعمال کیا ہے۔ یہاں ہم نے input type=text کا input box رکھا ہے جس سے اب یہ ایک input box کا لئے آج ہی وزٹ کریں:

نام دیا input اور اس کا سائز ہم نے 16 رکھا ہے۔

<br>

اس کے بعد ہم نے ایک br کا بیگ لگایا نئی لائن کے لئے جس سے ہمیں td کے اندر ایک اور لائن کا اضافہ مل جائے گا۔

</TD> </TR>

اس کے بعد ہم نے td اور tr کے بیگ کو بند کر دیا۔

<TR><TD>

اس اگلی لائن میں ہم نے پھر سے ایک tr بنایا اور اس کے اندر ایک td بنایا اب اس کے اندر ہم اپنے calculator کے پہلے چار ٹین بنائیں گے۔

```
<INPUT TYPE="button" NAME="one" VALUE= "1" OnClick="Calc.Input.value += '1'">
<INPUT TYPE="button" NAME="two" VALUE= "2" OnCLick="Calc.Input.value += '2'">
<INPUT TYPE="button" NAME="three" VALUE= "3" OnClick="Calc.Input.value += '3'">
<INPUT TYPE="button" NAME="plus" VALUE= "+" OnClick="Calc.Input.value += '+'">
```

یہاں ہم نے پہلے لائن کے لئے 4 ٹین بنانے میں اس کام کے لئے ہم نے چار input فیلڈ بنائی اور اس کا نام پر button رکھ دیا اس سے اب ہمارے پاس چار button بن گئے ہیں۔ پہلے والے کو ہم نے نام دیا one کا اور اس کی ویلودی 1 جو کہ ٹین کے اوپر شوہو گا اور ہمیں یا یوزر کو پتا چلے گا کہ یہ 1 نمبر کا ٹین ہے اس کے بعد ہم نے onclick کے attribute میں جو ویلودی ہے اس کے لئے ہم نے یہ form کا نام ہے اسے input جو کہ پہلی input فیلڈ کا نام ہے اس کے ساتھ concatenate کر دیا۔ اس سے Calc کو اس کے بعد ہم نے value کے ساتھ input کی ویلودی کے ساتھ ملا دے گا۔ اس سے ہمیں calculator میں لکھا جانے والا سب کچھ پہلی فیلڈ میں شوہو گا۔ اس کے بعد ہم نے Calc.Input کو JavaScript کے ایک کا نام value کے ساتھ concatenate کر دیا اس سے اب یہ اس ٹین کی ویلودی کو اٹھا کر Calc.Input کے ساتھ ملا دے گا اور پھر ہم نے اسے + اور اس کے بعد = کا سائن لگا کر 1 کی ویلودے دی جسے ہم نے اسٹرینگ کے طور پر لکھا ہے۔ + کے فوراً بعد = کے سائن کو سے ہم Assignment اور پیر کہتے ہیں۔ اس سے ویلودی میں پہلے ایک کا اضافہ ہو جاتا ہے۔ اس کے بعد ہم نے اسٹرینگ ویلودی 1 دی ہے ڈبل

کوٹس کے ساتھ 1 اسٹرینگ دیلو ہے۔

اس کو سمجھنے میں مشکل ہو سکتی ہے اسی لئے کوشش ضروری ہے۔ یہی عمل ہم نے باقی تینوں میں بھی دہرا�ا ہے۔

<br>

اس کے بعد ہم نے ایک br کا نیگ لگایا نئی لائن کے لئے جس سے ہمیں td کے اندر ایک اور لائن کا اضافہ مل جائے گا۔ جہاں ہم اپنے اگلے چار بٹن بنائیں گے۔

```
<INPUT TYPE="button" NAME="four" VALUE=" 4 " OnClick="Calc.Input.value += '4' ">
<INPUT TYPE="button" NAME="five" VALUE=" 5 " OnCLick="Calc.Input.value += '5' ">
<INPUT TYPE="button" NAME="six" VALUE=" 6 " OnClick="Calc.Input.value += '6' ">
<INPUT TYPE="button" NAME="minus" VALUE=" - " OnClick="Calc.Input.value += ' - ' ">
```

اس اگلی لائن میں ہم نے مزید چار بٹن بنائیں ہے اور یہاں بھی ہم نے وہی طریقہ کا استعمال کیا ہے جو پہلے چار بٹن بنانے کے لئے استعمال کیا تھا۔

<br>

اس کے بعد ہم نے پھر ایک br کا نیگ لگایا نئی لائن کے لئے جس سے ہمیں td کے اندر ایک اور لائن کا اضافہ مل جائے گا۔ اور اب ہم یہاں اپنے مزید چار بٹن بنائیں گے۔

```
<INPUT TYPE="button" NAME="seven" VALUE=" 7 " OnClick="Calc.Input.value += '7' ">
<INPUT TYPE="button" NAME="eight" VALUE=" 8 " OnCLick="Calc.Input.value += '8' ">
<INPUT TYPE="button" NAME="nine" VALUE=" 9 " OnClick="Calc.Input.value += '9' ">
<INPUT TYPE="button" NAME="times" VALUE=" x " OnClick="Calc.Input.value += ' * '">
```

اس لائن میں بھی ہم نے وہی طریقہ کار استعمال کیا ہے اور اس مرتبہ ہم نے آخری لائن میں ضرب کے لئے \* کا استعمال کیا ہے لیکن ویلوx کی دی ہے کیونکہ یہ ہمیں نظر آئے گا اور اس کا مزید کوئی کام نہیں ہے۔

<br>

اس کے بعد ہم نے ایک br کا ٹیگ لگایا نئی لائن کے لئے جس سے ہمیں td کے اندر ایک اور لائن کا اضافہ مل جائے گا۔ اور یہاں ہم اپنے آخری چار ٹین بنائیں کے اور اپنا کام کمکمل کریں گے۔

```
<INPUT TYPE="button" NAME="clear" VALUE=" c " OnClick="Calc.Input.value = ''">
<INPUT TYPE="button" NAME="zero"  VALUE=" 0 " OnClick="Calc.Input.value += '0' ">
<INPUT TYPE="button" NAME="Dolt"  VALUE=" = " OnClick="Calc.Input.value =
eval(Calc.Input.value)">
```

اس لائن میں ہم نے clear کے ٹین کو خالی رکھا ہے کیونکہ جب اس ٹین کو دبایا جائے گا تو سب پہنچتم ہو جانا چاہیے۔ اور زیر و کے لئے ویلوزیرا اور آخر میں doit کے نام سے ایک تیسرا ٹین جو کہ = کے لئے استعمال ہوگا۔ اور اسی کوڈ کے اندر ہم eval() کا استعمال کریں گے یہ math equations کو جمع کرنے کا کام کرتا ہے۔ اسی function کے اندر ہم Calc.input.value کو ایڈ کریں گے جس سے اب جتنے بھی ٹین دبائے پر ویلو ملے وہ اس کے اندر خود بے خود ایڈ ہو جائے گی۔ اور = کا ٹین دبائے پر ہمیں جواب مل جائے گا۔

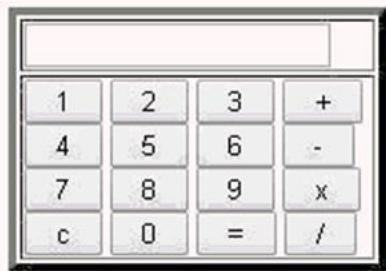
```
<INPUT TYPE="button" NAME="div"  VALUE="/" OnClick="Calc.Input.value += '/'">
```

اوپر ہم نے تین ٹین بنائے تھے لیکن ابھی چوتھا ٹین رہ گیا ہے اور وہ ہے تقسیم کا یعنی / کا اس کے لئے ہم نے آخری ٹین بھی اسی طریقہ کار سے تیار کیا اور پھر ہمیں تمام ٹیگز کو آخر میں بند کرنا ہے۔ یہاں یہ بتاتا چلوں کہ input کا ٹیگ بند نہیں ہوتا۔

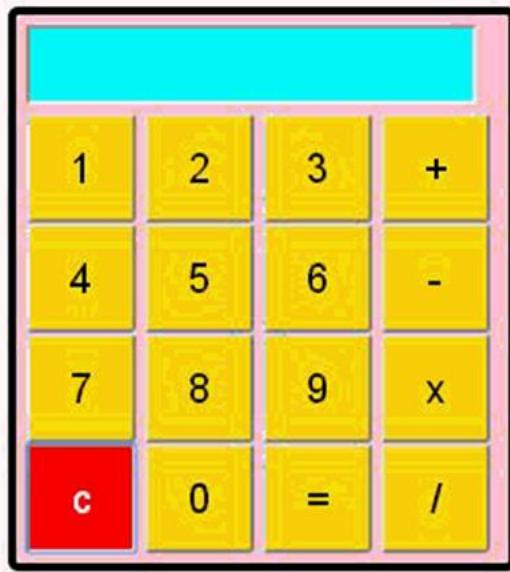
</TD></TR></TABLE></FORM>

اس کے بعد ہم نے آخر میں تمام ٹیگز form table tr td کو بند کر دیا۔ اب ہمارا Calculator تیار ہے۔

اس کا ریٹن یہ ہو گا۔



ہم اسے CSS کا استعمال کر کے مزید خوبصورت بھی بنائتے ہیں۔



### Temperature Convertor

ہم یہاں آپ کو ایک بنیادی Temperature Convertor بنانا سکھائیں گے جس میں ہم Fahrenheit اور Celsius کو ایک دوسرے سے Convert کریں گے۔ اس کا مکمل لئے ہم Math.round() کا استعمال کریں گے۔ لیکن اس سے پہلے ہم شروع کریں یہ ہمیں یہ پتا ہونا چاہئے ایک Celsius کا ہوتا ہے یا ایک Fahrenheit کا ہوتا ہے۔

1 Fahrenheit = -17.2222 Celsius

1 Celsius = 33.8 Fahrenheit

```
<input id="c" onkeyup="convert('C')">Celsius<br></br>
```

```
<input id="f" onkeyup="convert('F')">Fahrenheit
```

یہاں ہم نے دو input box بنائے ایک کو id کی ورنہ سرے کو f کی دونوں میں ہی ہم نے event کے onkeyup کا استعمال کیا ہے۔ اور دونوں میں convert کا نام سے جگہن بنایا ہے اس کا نام دے دیا۔ اور دونوں کو f اور c کے جو ویریبلز ہم نے convert کے فنگشن کے اندر بنائے ہیں وہ دے دئے۔

```
<script>

function convert(degree) {

    if (degree == "C") {

        F = document.getElementById("c").value * 9 / 5 + 32;

        document.getElementById("f").value = Math.round(F);

    } else

        C = (document.getElementById("f").value - 32) * 5 / 9;

        document.getElementById("c").value = Math.round(C);

    }

}

</script>
```

اب یہاں ہم نے script کے بیگ کے اندر ایک فنگشن بنایا convert کے نام سے اور اس کے اندر ایک undefined ویریبل بنایا degree کے نام سے پھر ہم نے فنگشن کے اندر ایک if کا فنگشن استعمال کیا اور اس کے اندر پو پرئی دی کہ اگر degree کا ویریبل

برابر کو C کے تو اور اس کو کنڈیشن دینے کے لئے ہم نے پہلے F کے نام سے ایک ویریبل بنایا اور اس کو یلو دے دی اور جو id میں فیلڈ میں وہ تھی C کے نام سے وہی دے دیا اور پھر ہم نے value کی کمانڈ کے ساتھ اسے concatenate کر دیا جس سے اب یہ اس میں سے یلو اٹھائے گا اور اسے بعد ہم نے 9/5+32 کو value سے ضرب کر دیا۔ اس طرح ضرب ہونے کے بعد جو یلو ملے گی وہ ہمیں کنورٹ شدہ یلو ملے گی۔ اگلی میں ہم نے پھر سے document.getElementById کا استعمال کیا اور اس کے اندر جو ہم نے دوسرے بٹن میں f کی id بنانی تھی اسے لکھا اور اسے بھی value کی کمانڈ کے ساتھ concatenate کر دیا پھر ہم نے ایک اور میٹھ اوبجیکٹ(); Math.round() کا استعمال کیا اس سے جو قریبی یلو ہوتی ہے وہ ہمیں ملتی ہے۔ اس کے اندر ہم نے F کے ویریبل کو ایڈ کر دیا۔

اس کے بعد ہم نے else کے لئے بھی وہی طریقہ کا استعمال کرتے ہوئے پہلے C کا ویریبل بنایا اس کے اندر ہم نے document.getElementById کا استعمال کیا لیکن اس بارہم نے اسے راؤنڈ بریکٹس کے اندر لکھا ہے کیوں کہ اس بارہم یہ تمام کی تمام یلو ضرب کرنی ہے۔ ہم نے document.getElementById کو جو ہم نے دوسرے بٹن میں id بنانی تھی f کے نام سے وہ دے دی اور value کی کمانڈ کے ساتھ concatenate کر دیا۔ اسے کے بعد 32 کو اگا کر راؤنڈ بریکٹ کو بند کیا اور اس کو ضرب کر دیا 9 / 5 سے اس کے بعد ہم نے پھر اس طریقہ ہی استعمال کرتے ہوئے document.getElementById کا استعمال کیا اور اس میں پہلے والی C کی id لکھ دی اور اس کو value کی کمانڈ کے ساتھ concatenate کر دیا اور اس میں بھی(); Math.round() کے اوبجیکٹ کا استعمال کیا لیکن اس بارہم نے F کے ویریبل کی جگہ C کے ویریبل کو ایک کر دیا۔

اب اس سے ہو گایہ کہ جب بھی کوئی یوزر پہلے input فیلڈ کے اندر کوئی دینا داخل کرے گا تو وہ یلو پہلے یا دوسرے جس میں یوزر یلو داخل کرتا ہے اس سے اٹھا کر پہلے ضرب پھر تقسیم ہو گی اور آخر میں چونکہ ہم نے Attribute onkeyup کا استعمال کیا ہے اس لئے onclick کے بر عکس یہ ہمیں فوراً ہی رزلٹ دے دیگا۔ اس کا رزلٹ یہ ہو گا۔

 Celsius

 Fahrenheit